Monoidal-Closed Categories of Tree Automata

Colin Riba
ENS de Lyon, Université de Lyon, LIP*
colin.riba@ens-lyon.fr
http://perso.ens-lyon.fr/colin.riba/

This paper surveys a new perspective on tree automata and Monadic Second-Order Logic (MSO) on infinite trees. We show that the operations on tree automata used in the translations of MSO-formulae to automata underlying Rabin's Tree Theorem (the decidability of MSO) correspond to the connectives of *Intuitionistic Multiplicative Exponential Linear Logic* (IMELL). Namely, we equip a variant of usual alternating tree automata (that we call *uniform* tree automata) with a fibred monoidal closed structure which in particular handles a linear complementation of alternating automata. Moreover, this monoidal structure is actually Cartesian on *non-deterministic* automata, and an adaptation of a usual construction for the simulation of alternating automata by non-deterministic ones satisfies the deduction rules of the $!(-)$ exponential modality of IMELL. (But this operation is unfortunately not a functor because it does not preserve composition.)

Our model of IMLL consists in categories of games which are based on usual categories of two-player linear sequential games called *simple games*, and which generalize usual acceptance games of tree automata. This model provides a realizability semantics, along the lines of Curry-Howard proofs-as-programs correspondence, of a linear constructive deduction system for tree automata. This realizability semantics, which can be summarized with the slogan "*automata as objects, strategies as morphisms*", satisfies an expected property of witness extraction from proofs of existential statements. Moreover, it makes it possible to combine realizers produced as interpretations of proofs with strategies witnessing (non-)emptiness of tree automata.

1. Introduction

Monadic Second-Order Logic (MSO) on infinite trees is a rich system, which contains non trivial mathematical theories (see e.g. [Rab69, BGG97]), and which subsumes many logics, in particular modal logics (see e.g. [BdRV02]) and logics for verification (see e.g. [VW08]). Rabin's Tree Theorem [Rab69], the decidability of MSO on infinite trees, is an "*important and difficult decidability theorem for mathematical theories*" ([BGG97, §1.3, p. 11]).

*Univ Lyon, EnsL, UCBL, LIP, F-69342, LYON Cedex 07, France

The original proof of [Rab69] relied on an effective translation of formulae to finite state automata running on infinite trees. Since then, there has been considerable work on Rabin's Tree Theorem, culminating in streamlined decidability proofs, as presented e.g. in [Tho97, GTW02, PP04]. Most current approaches to MSO on infinite trees are based on translations of MSO-formulae to automata.[1]

In this paper, we show that the operations on tree automata used in the translations of MSO-formulae to automata underlying Rabin's Tree Theorem correspond to the connectives of *Intuitionistic Multiplicative Exponential Linear Logic* (IMELL) [Gir87]. Namely, we equip a variant of usual alternating tree automata (that we call *uniform* tree automata) with a fibred monoidal closed structure which in particular (via determinacy of ω-regular games) handles a linear complementation of alternating automata. Moreover, this monoidal structure is actually Cartesian on *non-deterministic* automata, and an adaptation of a usual construction for the simulation of alternating automata by non-deterministic ones satisfies the deduction rules of the $!(-)$ exponential modality of IMELL. (But this operation is unfortunately not a functor because it does not preserve composition.)

Our model of IMLL consists in categories of games which are based on usual categories of two-player linear sequential games called *simple games* (see e.g. [Abr97, Hyl97]), and which generalize usual acceptance games of tree automata.[2] This model provides a realizability semantics, along the lines of the Curry-Howard proofs-as-programs correspondence (see e.g. [GLT89, SU06]), of a linear constructive deduction system for tree automata (see Fig. 1). This realizability semantics, which can be summarized with the slogan "*automata as objects, strategies as morphisms*", satisfies an expected property of witness extraction from proofs of existential statements. Moreover, it makes it possible to combine realizers obtained as interpretations of proofs with strategies witnessing (non-)emptiness of tree automata.

Our motivation for this deduction system is that even if Rabin's Tree Theorem proves the existence of decision procedures for MSO on infinite trees, there is (as far as we know) no working implementation of such procedures. The reason is that all known translations of formulae to tree automata involve at some stage the determinization of automata on ω-words (McNaughton's Theorem [McN66]), which is believed not to be amenable to tractable implementation (see e.g. [KV05]). We instead target semi-automatic approaches in which the user can interactively perform some proofs steps and can delegate sufficiently simple subgoals to automatic non-emptiness checkers (solving parity games). The partial proof tree built by the user is then translated to a combinator able to compose the witnessing strategies obtained from the algorithms.

This work builds on [Rib15], which proposed monoidal fibrations of games and tree automata, and extends it with a monoidal *closed* structure, based on a variant of alternating automata (that we call *uniform automata*), and which allows for a clearer connection of our model with IMELL. We follow the guidelines and axiomatizations provided by categorical logic and categorical approaches to the Curry-Howard correspondence, for which we refer to [Jac01, LS86] and [AC98]. We moreover refer to [Mel09] for a comprehensive presentation of categorical axiomatizations of models of (subsystems of) linear logic. In the remaining of this Introduction, we sketch some key points of Rabin's Theorem (§1.1, §1.2) and then outline the main aspects of our decomposition of MSO in IMELL and the corresponding realizability interpretation (§1.3–§1.5).

[1] But with the notable exception of [Blu13].

[2] However, the IMLL-structure underlying our model differs from the usual IMLL-structure of simple games.

1.1. MSO and (Non-Deterministic) Tree Automata. Let us set some concepts and notations. Concatenation of sequences s, t is denoted either $s.t$ or $s \cdot t$, and ϵ is the empty sequence. We fix throughout the paper a finite non-empty set $\mathfrak{D}$ of *tree directions*. We are interested in labelings of the full $\mathfrak{D}$-ary tree $\mathfrak{D}^*$ over different *alphabets*. Alphabets (denoted Σ, Γ, etc) are finite non-empty sets, and Σ-labeled $\mathfrak{D}$-ary trees are functions $T : \mathfrak{D}^* \to \Sigma$. Throughout the paper, we shall denote with overlines both vectors and finite words, so that e.g. $\overline{T}$ denotes a sequence $\overline{T} = T_1, \ldots, T_n$, while $\overline{\mathsf{a}} \in \Sigma^*$ denotes a word $\overline{\mathsf{a}} = \mathsf{a}_1. \cdots .\mathsf{a}_n$ where each a_i is a letter of Σ.

There are different expressively equivalent variants of MSO over infinite trees. The main idea is that we have a two-sorted logic, with a sort of *individuals* ranging over the positions of the full $\mathfrak{D}$-ary tree $\mathfrak{D}^*$ (that is over $\mathfrak{D}^*$ itself) and a sort of *monadic* second order variables ranging over sets of positions (that is over $\mathcal{P}(\mathfrak{D}^*)$). When discussing translations to automata, it is actually customary and convenient (following e.g. [Tho97]), to only allow monadic variables, and to simulate quantifications over individuals via a (definable) singleton predicate. We shall moreover not be concerned with any particular choice of atomic predicates. We thus assume given a set At of atomic predicates. MSO-formulae are then given by

$$\varphi, \psi \quad ::= \quad \alpha \quad | \quad \bot \quad | \quad \top \quad | \quad \neg\varphi \quad | \quad \varphi \wedge \psi \quad | \quad (\exists X)\varphi \qquad\qquad \text{(where } \alpha \in \text{At)}$$

These formulae are interpreted in the full $\mathfrak{D}$-ary tree $\mathfrak{D}^*$ as expected, assuming an interpretation of the atomic predicates.

On the other hand, there are two families of tree automata involved in the interpretation of MSO-formulae: *non-deterministic* tree automata and *alternating* tree automata[3]. The simplest notion is that of non-deterministic automaton, and it is sufficient to introduce the basic motivations and methodology of this work.

A tree automaton $\mathcal{A}$ consists of a finite set Q of states, with a distinguished[4] initial state $q^\imath \in Q$, an acceptance condition given by an ω-regular set $\Omega \subseteq Q^\omega$, and a transition function ∂. A *non-deterministic* tree automaton $\mathcal{A}$ over Σ has a transition function of the form

$$\partial \quad : \quad Q \times \Sigma \quad \longrightarrow \quad \mathcal{P}(\mathfrak{D} \longrightarrow Q)$$

Acceptance for tree automata can equivalently be described by *games* or *run trees*. The notion of run tree is simpler and sufficient at various places in this Introduction and §2. A *run tree* of $\mathcal{A}$ on $T : \mathfrak{D}^* \to \Sigma$ is a tree $R : \mathfrak{D}^* \to Q$ such that $R(\epsilon) = q^\imath$, and which respects the transitions of $\mathcal{A}$, in the sense that for each tree position $p \in \mathfrak{D}^*$, there exists a $\mathfrak{D}$-tuple $(q_d)_{d \in \mathfrak{D}} \in \partial(R(p), T(p))$ such that $R(p.d) = q_d$ for all $d \in \mathfrak{D}$. The run R is *accepting* if all its infinite paths belong to Ω. We say that T is accepted by $\mathcal{A}$ if there exists an accepting run of $\mathcal{A}$ on T, and let $\mathcal{L}(\mathcal{A})$ be the set of trees accepted by $\mathcal{A}$. We moreover write $\mathcal{A}(T)$ for the set of accepting runs of $\mathcal{A}$ on T.

1.2. Games and Alternating Automata. The main difficulty when translating MSO-formulae to tree automata is the interplay between negation and (existential) quantification. Historically, Rabin [Rab69] translated MSO-formulae to *non-deterministic* tree automata. The major achievement of Rabin [Rab69] was to show that non-deterministic automata on infinite trees are closed under complement. This means that for every non-deterministic automaton $\mathcal{A}$ one can build a non-deterministic automaton $\sim\mathcal{A}$ which accepts exactly the trees rejected by $\mathcal{A}$.

Rabin's original construction [Rab69] of a complement $\sim\mathcal{A}$ from $\mathcal{A}$ has been considerably simplified by Gurevich and Harrington [GH82] thanks to the notion of *acceptance game*. The

[3] Alternating automata are not always made explicit (see e.g. [Tho97]).

[4] It is also customary (and equivalent in terms of expressiveness) to allow several initial states.

idea is to model the evaluation of an automaton $\mathcal{A}$ on an input tree T as an infinite two-players game $\mathcal{G}(\mathcal{A}, T)$. In this game, the *Proponent* P (also called ∃loïse or Automaton) plays for acceptance while its *Opponent* O (also called ∀bélard or Pathfinder) plays for rejection, and $\mathcal{A}$ accepts T when P has a winning strategy. A typical (infinite) play χ in $\mathcal{G}(\mathcal{A}, T)$ has the form:

$$\begin{array}{ccccccccccccc}
\mathsf{P} & & \mathsf{O} & & \mathsf{P} & & \mathsf{O} & & & & \mathsf{P} & & \mathsf{O} \\
(q_{0,d})_{d\in\mathfrak{D}} & \cdot & d_0 & \cdot & (q_{1,d})_{d\in\mathfrak{D}} & \cdot & d_1 & \cdot & \cdots & \cdot & (q_{n+1,d})_{d\in\mathfrak{D}} & \cdot & d_{n+1} & \cdot & \cdots \\
\pitchfork & & \pitchfork & & \pitchfork & & \pitchfork & & & & \pitchfork & & \pitchfork & & \\
\partial(q^\iota, T(\epsilon)) & & \mathfrak{D} & & \partial(q_{0,d_0}, T(d_0)) & & \mathfrak{D} & & & & \partial(q_{n,d_n}, T(p)) & & \mathfrak{D} & &
\end{array}$$

where $p = d_0 \cdot \ldots \cdot d_n$. Then χ is winning for P if the sequence of states $q^\iota, q_{0,d_0}, q_{1,d_1}, \ldots$ belongs to Ω; otherwise it is winning for O. Note that P chooses transitions $(q_d)_{d\in\mathfrak{D}}$ while O chooses tree directions $d \in \mathfrak{D}$. Hence, there is a bijection between accepting runs $R \in \mathcal{A}(T)$ and winning P-strategies in $\mathcal{G}(\mathcal{A}, T)$. Since acceptance games are determined, $\mathcal{A}$ does not accept T precisely when O has a winning strategy in $\mathcal{G}(\mathcal{A}, T)$. Gurevich and Harrington [GH82] show that in acceptance games, winning strategies can be assumed to be finite state w.r.t. game positions of the form $(p, q) \in \mathfrak{D}^* \times Q$, that is to only depend on a finite memory in addition to the game positions in $\mathfrak{D}^* \times Q$.[5] This makes it possible to devise an automaton $\sim\mathcal{A}$ which, using a usual projection operation, non-deterministically checks for the existence of winning O-strategies.

However, the construction of $\sim\mathcal{A}$ is still not trivial because the roles of P and O in acceptance games are not symmetric, so that dualizing the acceptance game of a non-deterministic automaton $\mathcal{A}$ does not directly give a *non-deterministic* automaton $\sim\mathcal{A}$. Since [MS87, EJ91, MS95] it is known that the construction of $\sim\mathcal{A}$ can be neatly decomposed using *alternating* automata. The original idea, as stated in e.g. [MS87, MS95], is for an alternating automaton $\mathcal{A}$ with state set Q to have transitions with values in the free distributive lattice over $Q \times \mathfrak{D}$. But recall from e.g. [Joh86, Lem. I.4.8] that free distributive lattices are given by irredundant disjunctive normal forms. Actually, following [Wal02], we can give up irredundancy. We thus simply assume that transitions are of the form

$$\partial \quad : \quad Q \times \Sigma \quad \longrightarrow \quad \mathcal{P}(\mathcal{P}(Q \times \mathfrak{D})) \tag{1}$$

and we read $\partial(q, \mathsf{a})$ as the disjunctive normal form

$$\bigvee_{\gamma\in\partial(q,\mathsf{a})} \bigwedge_{(q',d)\in\gamma} (q', d)$$

This results in acceptance games where intuitively P plays from disjunctions while O plays from conjunctions. A typical play in the acceptance game $\mathcal{G}(\mathcal{A}, T)$ with $\mathcal{A}$ alternating has the form

$$\begin{array}{ccccccccccc}
\mathsf{P} & & \mathsf{O} & & \mathsf{P} & & \mathsf{O} & & & & \mathsf{P} & & \\
\gamma_0 & \cdot & (q_0, d_0) & \cdot & \gamma_1 & \cdot & (q_1, d_1) & \cdot & \cdots & \cdot & \gamma_{n+1} & \cdot & \cdots \\
\pitchfork & & \pitchfork & & \pitchfork & & \pitchfork & & & & \pitchfork & & \\
\partial(q^\iota, T(\epsilon)) & & \gamma_0 & & \partial(q_0, T(d_0)) & & \gamma_1 & & & & \partial(q_n, T(p)) & &
\end{array}$$

Hence, P chooses relations $\gamma_k \in \mathcal{P}(Q \times \mathfrak{D})$ instead of tuples $(q_{k,d})_{d\in\mathfrak{D}}$ while O chooses pairs $(q_k, d_k) \in \gamma_k$ instead of just tree directions $d_k \in \mathfrak{D}$. The main consequence is that O may now be allowed to choose between pairs $(q'_k, d_k), (q''_k, d_k) \in \gamma_k$ with *different states* q'_k, q''_k for the same tree direction $d_k \in \mathfrak{D}$.

[5] This is trivial for P-strategies but not for O-strategies.

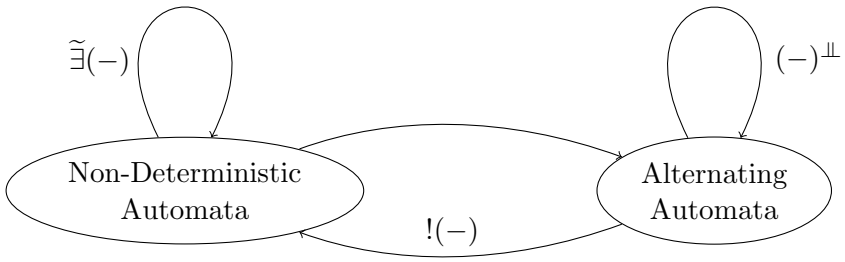
The extra possibility for O to choose states in addition to tree directions allows us to define a complement of $\mathcal{A}$ which essentially simulates $\mathcal{A}$ while reversing the roles of P and O. This can be implemented with an alternating automaton[6] $\mathcal{A}^{\perp\!\!\!\perp}$ having the same states as $\mathcal{A}$. The idea is that since the double powerset $\mathcal{P}(\mathcal{P}(Q \times \mathfrak{D}))$ in (1) represents disjunctive normal forms over $Q \times \mathfrak{D}$, the transition function of $\mathcal{A}^{\perp\!\!\!\perp}$ can just take $(q, \mathtt{a}) \in Q \times \Sigma$ to a disjunctive normal form representing the dual of $\partial(q, \mathtt{a})$. Then, if the acceptance condition of $\mathcal{A}^{\perp\!\!\!\perp}$ is the complement of Ω, it follows from game determinacy that $\mathcal{L}(\mathcal{A}^{\perp\!\!\!\perp})$ is the complement of $\mathcal{L}(\mathcal{A})$.

Every alternating automaton $\mathcal{A}$ can be simulated by a non-deterministic automaton $!\mathcal{A}$ of exponential size (this is the *Simulation Theorem* [MS87, EJ91, MS95], see also §7.2), while non-deterministic automata are linearly embedded into alternating automata via the obvious mapping

$$(q_d)_{d \in \mathfrak{D}} \in Q^{\mathfrak{D}} \quad \longmapsto \quad \{(q_d, d) \mid d \in \mathfrak{D}\} \in \mathcal{P}(Q \times \mathfrak{D})$$

On the other hand, non-deterministic automata are easily (and linearly in the number of states) closed under projections $\widetilde{\exists}_\Sigma(-)$ which implement the existential quantifications of MSO.

The situation can be pictured as follows:



(2)

Accordingly, in most modern approaches to MSO on infinite trees, the complementation of non-deterministic tree automata can be decomposed as

$$\sim\mathcal{A} \quad = \quad !(\mathcal{A}^{\perp\!\!\!\perp}) \tag{3}$$

1.3. Toward Linear Logic. The model of [Rib15] consists in categories of two-player sequential games generalizing the usual acceptance games of tree automata. Using the notion of *uniform automata* (to be introduced in §3), the extension of [Rib15] proposed in this work shows that the decomposition depicted in (2) of the translation of MSO-formulae to non-deterministic tree automata via alternating automata corresponds to some extent to an IMELL-structure:

- First, the usual direct synchronous product of alternating automata (which we denote $(-) \otimes (-)$) has a symmetric monoidal structure. Moreover, thanks to the monoidal-*closed* structure of $(-) \otimes (-)$ on uniform automata, the set of morphisms from $\mathcal{A}$ to $\mathcal{B}$ is in bijection with the set of winning P-strategies in the acceptance game of an automaton $(\mathcal{A} \multimap \mathcal{B})$ over T. In particular, linear complements are obtained with

  $$\mathcal{A}^{\perp\!\!\!\perp} \quad \simeq \quad \mathcal{A} \multimap \perp\!\!\!\perp$$

  (where $\perp\!\!\!\perp$ is a particular automaton accepting no tree), with as expected $T \in \mathcal{L}(\mathcal{A}^{\perp\!\!\!\perp})$ iff $T \notin \mathcal{L}(\mathcal{A})$.

[6] $(-)^{\perp\!\!\!\perp}$ was noted $\sim(-)$ in [Rib15].

- Second, we show that the simulation operation $!(-)$ satisfies the *deduction rules* of the usual modality $!(-)$ of IMELL. Moreover, the symmetric monoidal product $(-) \otimes (-)$ is Cartesian on non-deterministic automata, so that the picture (2) is similar to the usual linear-non-linear adjunctions in models of IMELL. (Unfortunately, in our models the operation $!(-)$ is not a functor.[7])

The connection between alternating automata and IMELL suggests that we may take variants of IMELL as intermediate steps between MSO and automata. In our setting, an IMELL-based language for MSO would consist of the following formulae:

$$\varphi, \psi \quad ::= \quad \alpha \quad | \quad \bot \quad | \quad \mathbf{I} \quad | \quad \varphi \otimes \psi \quad | \quad \varphi \multimap \psi \quad | \quad !\varphi \quad | \quad (\exists X)\varphi \quad | \quad (\forall X)\varphi$$

This language must be seen as a refinement of MSO with finer-grained connectives, which directly correspond to operations on automata (the primitive universal quantification is actually *non-standard*, see §6). Since the connectives of IMELL correspond to operations on automata, provided a (non-deterministic) automaton $\mathcal{A}(\alpha)$ is given for each atomic formula $\alpha \in \mathrm{At}$, one can (inductively) associate an automaton $\mathcal{A}(\varphi)$ to each IMELL formula φ.

It would have been natural to also consider the *additive* connectives & (conjunction) and $\oplus$ (disjunction) of linear logic, which do correspond to known constructions on alternating automata. However, the expected categorical properties of these connectives would require an extension of our setting that we leave for further work. Keeping this in mind, the translation of MSO to non-deterministic automata induced by (3) factors via the map $(-)^{\mathrm{nd}} : \mathsf{MSO} \to \mathsf{IMELL}$ given by

$$\begin{array}{rcl} \alpha^{\mathrm{nd}} & := & \alpha \end{array} \qquad \begin{array}{rcl} \bot^{\mathrm{nd}} & := & \bot \\ (\neg\varphi)^{\mathrm{nd}} & := & !(\varphi^{\mathrm{nd}} \multimap \bot) \\ (\varphi \wedge \psi)^{\mathrm{nd}} & := & \varphi^{\mathrm{nd}} \otimes \psi^{\mathrm{nd}} \\ ((\exists X)\varphi)^{\mathrm{nd}} & := & (\exists X)\varphi^{\mathrm{nd}} \end{array} \qquad \begin{array}{rcl} \top^{\mathrm{nd}} & := & \mathbf{I} \end{array}$$

while the translation of MSO to alternating automata factors via the map $(-)^{\mathrm{alt}} : \mathsf{MSO} \to \mathsf{IMELL}$ given by

$$\begin{array}{rcl} \alpha^{\mathrm{alt}} & := & \alpha \end{array} \qquad \begin{array}{rcl} \bot^{\mathrm{alt}} & := & \bot \\ (\neg\varphi)^{\mathrm{alt}} & := & \varphi^{\mathrm{alt}} \multimap \bot \\ (\varphi \wedge \psi)^{\mathrm{alt}} & := & \varphi^{\mathrm{alt}} \otimes \psi^{\mathrm{alt}} \\ ((\exists X)\varphi)^{\mathrm{alt}} & := & (\exists X)!\varphi^{\mathrm{alt}} \end{array} \qquad \begin{array}{rcl} \top^{\mathrm{alt}} & := & \mathbf{I} \end{array}$$

The factorizations of the translations of MSO to automata via IMELL are sound in the following sense.

Proposition 1.1. *Let $(-)^{\dagger}$ be either $(-)^{\mathrm{nd}}$ or $(-)^{\mathrm{alt}}$. A closed MSO-formulae φ is true in the full infinite $\mathfrak{D}$-ary tree if and only if $\mathcal{A}(\varphi^{\dagger})$ accepts the unique $\mathbf{1}$-labeled tree.*

1.4. Computational Interpretation of Proofs. In our view, proposing IMELL as an intermediate system between MSO and automata should rely on a suitable computational interpretation of proofs, along the lines of the Curry-Howard proofs-as-programs correspondence. We explain here our view that the relevant computational objects are runs of automata or P-strategies in acceptance games. This leads us to the slogan "*automata as objects, strategies as morphisms*", and implies that we consider a deduction system for *automata* rather than IMELL formulae.

[7] It does not preserve composition, because of issues with positionality of strategies. Possible workarounds, left as future work, are discussed in §8.1.

Our deduction system manipulates sequents of the form

$$T \; ; \; \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B} \tag{4}$$

where T is an infinite tree labeled over (say) the alphabet Σ, and $\mathcal{A}_1, \ldots, \mathcal{A}_n, \mathcal{B}$ are tree automata over Σ. We see these sequents with two different levels of interpretation. The first level interprets *provability*: if the sequent (4) is provable, then the automaton $\mathcal{B}$ accepts the tree T as soon as the automata $\mathcal{A}_1, \ldots, \mathcal{A}_n$ all accept T.

The second level is the *computational* interpretation of *proofs* of the Curry-Howard correspondence. This is best exemplified with existential quantifications. The existential quantifications of MSO are implemented by a *projection* operation on non-deterministic automata. Consider a non-deterministic automaton $\mathcal{A}$ over the alphabet $\Gamma \times \Sigma$. Its projection $\widetilde{\exists}_\Sigma \mathcal{A}$ is the non-deterministic automaton over Γ defined as $\mathcal{A}$ but with transition function

$$\begin{array}{rccc} \partial_{\widetilde{\exists}_\Sigma \mathcal{A}} & : & Q_\mathcal{A} \times \Gamma & \longrightarrow & \mathcal{P}(\mathfrak{D} \to Q_\mathcal{A}) \\ & & (q, \mathtt{b}) & \longmapsto & \bigcup_{\mathtt{a} \in \Sigma} \partial_\mathcal{A}(q, (\mathtt{b}, \mathtt{a})) \end{array}$$

As expected, $\widetilde{\exists}_\Sigma \mathcal{A}$ accepts $T : \mathfrak{D}^* \to \Gamma$ iff there exists $U : \mathfrak{D}^* \to \Sigma$ such that $\mathcal{A}$ accepts $\langle T, U \rangle : \mathfrak{D}^* \to \Gamma \times \Sigma$.

Consider now a non-deterministic automaton $\mathcal{B}$ over the alphabet $\Sigma \simeq \mathbf{1} \times \Sigma$, where $\mathbf{1} \simeq \{\bullet\}$ is a singleton set. By *computational interpretation of proofs*, we mean that from a formal proof of the sequent

$$\mathbf{1} \; ; \vdash \widetilde{\exists}_\Sigma \mathcal{B}$$

(where $\mathbf{1}$ stands for the unique $\mathbf{1}$-labeled tree) one should be able to extract a witness for the existential quantification $\widetilde{\exists}_\Sigma \mathcal{B}$, that is a Σ-labeled tree accepted by $\mathcal{B}$. Such witnesses can actually be extracted from the runs of $\widetilde{\exists}_\Sigma \mathcal{B}$ on $\mathbf{1}$. First note that a run R of a non-deterministic automaton $\mathcal{A}$ on T defines a function $p \in \mathfrak{D}^* \longmapsto (q_d)_{q \in \mathfrak{D}} \in \partial_\mathcal{A}(R(p), T(p))$. It follows that given an accepting run R of $\widetilde{\exists}_\Sigma \mathcal{B}$ on $\mathbf{1}$, then from the induced function

$$p \in \mathfrak{D}^* \quad \longmapsto \quad (q_d)_{d \in \mathfrak{D}} \in \bigcup_{\mathtt{a} \in \Sigma} \partial_\mathcal{B}(R(p), \mathtt{a})$$

one can get a Σ-labeled tree T such that R is an accepting run of $\mathcal{B}$ on T.

In other words, *runs* of automata convey the kind of information one is usually interested in with computational interpretations of proofs. We will however rather rely on the more complex notions of acceptance games and strategies. There are two reasons for this choice. First, as discussed in §1.2 above, games give a smooth treatment of complementation of tree automata. The second reason, which we motivate with more details in §2, is that games and strategies are equipped with well-known categorical structures, which allow to easily define compositional interpretations of proofs.

Following the methodology of categorical logic, the categories proposed here and in [Rib15] are *indexed* (or *fibred*) over a base category $\mathbf{T}$ of trees, whose objects are alphabets and whose morphisms from Σ to Γ induce functions from Σ-labeled trees to Γ-labeled trees (see §2.2 and §4). In this setting, existential quantifications (in the categorical sense) are provided by a slight modification (denoted $\exists(-)$) of the usual projection $\widetilde{\exists}(-)$.

1.5. Toward Realizability Interpretations of MSO. The ultimate motivation for the Curry-Howard approach to automata on infinite trees proposed in this paper, together with the underlying decomposition of the translation of MSO-formulae to tree automata via IMELL, is to

provide realizability interpretations of MSO (in the spirit of e.g. [SU06, Koh08]). We think that the model presented here (consolidating [Rib15]) is a preliminary step toward this goal. Let us briefly describe our main results in this direction.

Generalizing (4), our deduction system also manipulates sequents of the form

$$M \; ; \; \mathcal{A}_1, \dots, \mathcal{A}_n \vdash \mathcal{B} \tag{5}$$

(see §2.2) where M is a $\mathbf{T}$-morphism, from say Σ to Γ and the automata $\mathcal{A}_1, \dots, \mathcal{A}_n, \mathcal{B}$ have input alphabet Γ. In case M is the identity $\mathbf{T}$-map on Σ, the sequent (5) is written

$$\Sigma \; ; \; \mathcal{A}_1, \dots, \mathcal{A}_n \vdash \mathcal{B} \tag{6}$$

which in contrast with (4) and (5) does not mention any tree. The full system is presented in Fig. 1, and we can state a second soundness result.

Proposition 1.2. *Given* IMELL*-formulae $\varphi_1, \dots, \varphi_n, \varphi$, we have*

$$\varphi_1, \dots, \varphi_n \vdash_{\mathsf{IMELL}} \varphi \qquad \Longrightarrow \qquad \mathbf{2}^p \; ; \; \mathcal{A}(\varphi_1), \dots, \mathcal{A}(\varphi_n) \vdash \mathcal{A}(\varphi)$$

(where $\varphi_1, \dots, \varphi_n, \varphi$ have free variables among $X_1, \dots, X_p$).

The symmetric monoidal closed structure, together with the categorical quantifiers and the interpretation of simulation as an exponential modality $!(-)$, allows us to interpret proofs in the deduction system of Fig. 1. From a proof $\mathscr{D}$ of a sequent (6), one can (compositionally w.r.t. the structure of $\mathscr{D}$) extract a winning finite-state strategy σ in an infinite game of the form $\mathcal{A}_1 \otimes \dots \otimes \mathcal{A}_n \multimap \mathcal{B}$. As indicated in §1.3, $\mathcal{A}_1 \otimes \dots \otimes \mathcal{A}_n$ essentially evaluates the automata $\mathcal{A}_1, \dots, \mathcal{A}_n$ in parallel, while the linear arrow $\multimap$ is a synchronous restriction of the usual linear arrow of simple games. When we have a strategy σ winning in $\mathcal{A}_1 \otimes \dots \otimes \mathcal{A}_n \multimap \mathcal{B}$, we say that σ is a *realizer* and write

$$\sigma \quad \Vdash \quad \mathcal{A}_1 \otimes \dots \otimes \mathcal{A}_n \quad \longrightarrow \quad \mathcal{B}$$

In case (6) is of the form $\mathbf{1} \; ; \vdash \exists_\Sigma \mathcal{N}$ (with $\mathcal{N}$ non-deterministic), we indeed obtain a computational interpretation of proofs in the sense of §1.4, since as shown in §7.1.2, we have

$$\sigma \Vdash \exists_\Sigma \mathcal{N} \qquad \Longleftrightarrow \qquad \sigma = \langle T, \tau \rangle \quad \text{where} \quad T : \mathfrak{D}^* \to \Sigma \quad \text{and} \quad \tau \Vdash \mathcal{N}(T)$$

Assume now that (6) is of the form

$$\Sigma \; ; \; \mathcal{A} \vdash \mathcal{B} \tag{7}$$

Then a Σ-labeled tree T induces a *substitution functor* $T^\star$, whose action on σ gives a function $T^\star(\sigma)$ taking any winning P-strategy τ on $\mathcal{A}(T)$ to a winning P-strategy $T^\star(\sigma) \circ \tau$ on $\mathcal{B}(T)$ (see Prop. 4.11). This gives the rule

$$\frac{\sigma \Vdash \mathcal{A} \multimap \mathcal{B}}{T^\star(\sigma) \Vdash \mathcal{A}(T) \multimap \mathcal{B}(T)}$$

and the function

$$\tau \Vdash \mathcal{A}(T) \qquad \longmapsto \qquad T^\star(\sigma) \circ \tau \Vdash \mathcal{B}(T)$$

In other words, realizers of sequents of the form (7) can be composed (via substitution) with strategies τ on $\mathcal{A}(T)$ obtained by any possible mean.

To summarize, we get the following soundness result.

Theorem 1.3. *Given automata $\overline{\mathcal{A}}, \mathcal{A}$ over Σ,*

$$\Sigma \; ; \; \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{A} \qquad \Longrightarrow \qquad \Vdash \mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{A} \tag{8}$$

$$\Vdash \mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{A} \qquad \Longrightarrow \qquad \mathcal{L}(\mathcal{A}_1) \cap \cdots \cap \mathcal{L}(\mathcal{A}_n) \subseteq \mathcal{L}(\mathcal{A}) \tag{9}$$

More generally, the methodology behind our deduction system and its realizability interpretation targets interactive proofs systems, allowing possible human simplifications or decompositions of the goals given to automatic tools, and moreover to *combine* the corresponding witnessing strategies. Our motivation is that even if Rabin's Tree Theorem proves the existence of decision procedures for MSO on infinite trees, there is (as far as we know) no working implementation of such procedures. The reason is that all known translations of formulae to tree automata involve at some stage the determinization of automata on ω-words (McNaughton's Theorem [McN66]), which is believed not to be amenable to tractable implementation (see e.g. [KV05]). We instead target semi-automatic approaches in which the user can delegate sufficiently simple subgoals to automatic non-emptiness checkers (solving parity games). The partial proof tree built by the user is then translated to a combinator able to compose the strategies obtained by the algorithms.[8] To this end, some relevant properties of our framework are the following.

First, thanks to the (non-standard) primitive universal quantifications, games of the form $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$ are equivalent to games of the form $\mathbf{1} \vdash \forall_\Sigma(\mathcal{A} \multimap \mathcal{B})$, which are themselves equivalent to ω-regular games on finite graphs. Thanks to the Büchi-Landweber Theorem [BL69], one can thus decide if there is a strategy realizing the implication $\mathcal{A} \multimap \mathcal{B}$, and if such a strategy exists, then there exists a finite state one, which is moreover effectively computable from $\mathcal{A}$ and $\mathcal{B}$ (see Cor. 6.5).

Second, following Ex. 6.10, our system can be extended with the rule

$$\frac{\mathcal{L}(\mathcal{A} : \mathbf{1}) \neq \emptyset}{\mathbf{1} \; ; \vdash \mathcal{A}}$$

This rule has the following consequences:

(1) Assuming

$$\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \quad \multimap \quad \mathcal{B}$$

(over say Σ) is realizable, following the same reasoning as for Cor. 6.5, we get (leaving implicit some structural and cut rules)

$$\cfrac{\cfrac{\cfrac{\cfrac{\mathcal{L}(\forall_\Sigma(\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B})) \neq \emptyset}{\mathbf{1} \; ; \vdash \forall_\Sigma(\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B})}}{\Sigma \; ; \vdash \mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B}}}{\Sigma \; ; \; \mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \vdash \mathcal{B}}}{\Sigma \; ; \; \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B}}$$

This entails the rules of Ex. 5.9, and in particular allows us to derive the general (Weak) rule

$$\frac{M \; ; \; \overline{\mathcal{A}}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M \; ; \; \overline{\mathcal{A}}, \mathcal{A}, \overline{\mathcal{B}} \vdash \mathcal{C}}$$

[8] The author and P. Pradic have recently obtained preliminary results in this direction for MSO on ω-words [PR17, PR18].

(2) Given $\mathcal{A}, \mathcal{B} : \Sigma$ non-deterministic we have

$$\frac{\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B}) = \emptyset}{\Sigma \; ; \; \mathcal{A} \vdash \mathcal{B}^{\perp\!\!\!\perp}}$$

Indeed, from $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B}) = \emptyset$ we can derive (again leaving implicit some structural and (Cut) rules)

$$\begin{array}{c}
\mathcal{L}\big(\exists_\Sigma(\mathcal{A} \otimes \mathcal{B}) \multimap \perp\!\!\!\perp\big) \neq \emptyset \\
\hline
\mathbf{1} \; ; \; \vdash \exists_\Sigma(\mathcal{A} \otimes \mathcal{B}) \multimap \perp\!\!\!\perp \\
\hline
\mathbf{1} \; ; \; \exists_\Sigma(\mathcal{A} \otimes \mathcal{B}) \vdash \perp\!\!\!\perp \\
\hline
\Sigma \; ; \; \mathcal{A} \otimes \mathcal{B} \vdash \perp\!\!\!\perp \\
\hline
\Sigma \; ; \; \mathcal{A}, \mathcal{B} \vdash \perp\!\!\!\perp \\
\hline
\Sigma \; ; \; \mathcal{A} \vdash \mathcal{B}^{\perp\!\!\!\perp}
\end{array}$$

Moreover any (finite-state) O-strategy witnessing $\mathcal{L}(\mathcal{A} \otimes \mathcal{B}) = \emptyset$ can be lifted to a (finite-state) realizer of $\mathcal{A} \multimap \mathcal{B}^{\perp\!\!\!\perp}$ (Prop. 7.7).

(3) In particular, for $\mathcal{A}, \mathcal{B} : \Sigma$ not-necessarily non-deterministic, we have

$$\frac{\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})}{\Sigma \; ; \; !\mathcal{A} \vdash ?\mathcal{B}}$$

where $?(-) := !((-) \multimap \perp\!\!\!\perp) \multimap \perp\!\!\!\perp$ (Prop. 7.17).

1.6. Outline. The paper is organized as follows. In §2, we expose some ingredients and methodology of our approach based on categorical logic, and we sketch the connection between IMELL and the interpretation of MSO in usual tree automata. We then turn in §3 to our notion of *uniform automata* (motivated by monoidal closure), and present basic material on zig-zag games required for our setting. Section 4 then deals with the fibred structure (which is essentially a refinement of [Rib15]), §5 presents the monoidal closure and the corresponding deduction rules, while §6 deals with quantifications. Finally, in §7 we concentrate on the Cartesian structure of non-deterministic automata and present the interpretation of the Simulation Theorem using the deduction rules of usual $!(-)$ IMELL-exponential modalities. App. A provides connections with usual game semantics. Further examples, showing that our setting can handle constructions of [CL08, SA05], are presented in App. C.

2. Toward Categories of Games and Automata

The purpose of this Section is twofold. First, in §2.1–2.3, we expose some ingredients and methodology of our approach based on categorical logic. We state in §2.1 the minimal requirements imposed by categorical semantics of proofs, and §2.2 presents some basic ideas and motivations on indexed categories for modeling free variables and quantifications. Finally, in §2.3 we explain why it seems difficult to obtain a suitable categorical semantics of implications using usual connectives on automata. Second, building on [Rib15], in §2.4 & §2.5 we sketch the connection between linear logic and the interpretation of MSO in tree automata mentioned in §1.3.

$$(\textsc{Exchange})\ \frac{M\ ;\ \overline{\mathcal{A}},\ \mathcal{A},\ \mathcal{B},\ \overline{\mathcal{C}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\ \mathcal{B},\ \mathcal{A},\ \overline{\mathcal{C}} \vdash \mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A}}{M \circ M'\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A}}\ (\textsc{Subst})$$

$$(\textsc{Cut})\ \frac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\ ;\ \overline{\mathcal{B}},\ \mathcal{A},\ \overline{\mathcal{C}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{B}},\ \overline{\mathcal{A}},\ \overline{\mathcal{C}} \vdash \mathcal{C}} \qquad \frac{}{M\ ;\ \mathcal{A} \vdash \mathcal{A}}\ (\textsc{Axiom})$$

$$(\textsc{Left}\ \otimes)\ \frac{M\ ;\ \overline{\mathcal{A}},\ \mathcal{A},\ \mathcal{B},\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\ \mathcal{A} \otimes \mathcal{B},\ \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\ ;\ \overline{\mathcal{B}} \vdash \mathcal{B}}{M\ ;\ \overline{\mathcal{A}},\ \overline{\mathcal{B}} \vdash \mathcal{A} \otimes \mathcal{B}}\ (\textsc{Right}\ \otimes)$$

$$(\textsc{Left}\ \mathbf{I})\ \frac{M\ ;\ \overline{\mathcal{A}},\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\ \mathbf{I},\ \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{}{M\ ;\ \vdash \mathbf{I}}\ (\textsc{Right}\ \mathbf{I})$$

$$(\textsc{Left}\ \multimap)\ \frac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\ ;\ \overline{\mathcal{B}},\ \mathcal{B},\ \overline{\mathcal{C}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{B}},\ \overline{\mathcal{A}},\ \mathcal{A} \multimap \mathcal{B},\ \overline{\mathcal{C}} \vdash \mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{A}},\ \mathcal{B} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{B} \multimap \mathcal{C}}\ (\textsc{Right}\ \multimap)$$

$$(\textsc{Dereliction})\ \frac{M\ ;\ \overline{\mathcal{A}}\ ,\ \mathcal{A}\ ,\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}}\ ,\ !\mathcal{A}\ ,\ \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{N}} \vdash \mathcal{A}}{M\ ;\ \overline{\mathcal{N}} \vdash !\mathcal{A}}\ (\textsc{Promotion})$$

$$(\textsc{Weak}_{\textsc{nd}})\ \frac{M\ ;\ \overline{\mathcal{A}}\ ,\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}}\ ,\ \mathcal{N}\ ,\ \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{A}}\ ,\ \mathcal{N}\ ,\ \mathcal{N}\ ,\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}}\ ,\ \mathcal{N}\ ,\ \overline{\mathcal{B}} \vdash \mathcal{C}}\ (\textsc{Contr}_{\textsc{nd}})$$

$$(\textsc{Left}\ \exists)\ \frac{M \times \mathrm{Id}_\Gamma\ ;\ \overline{\mathcal{A}[\pi]},\ \mathcal{B} \vdash \mathcal{A}[\pi]}{M\ ;\ \overline{\mathcal{A}},\ \exists_\Gamma \mathcal{B} \vdash \mathcal{A}} \qquad \frac{M \times N\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A}}{M \times N\ ;\ \overline{\mathcal{A}} \vdash (\exists_\Gamma \mathcal{A})[\pi]}\ (\textsc{Right}\ \exists)$$

$$(\textsc{Left}\ \forall)\ \frac{M \times N\ ;\ \overline{\mathcal{A}},\ \mathcal{B} \vdash \mathcal{A}}{M \times N\ ;\ \overline{\mathcal{A}},\ (\forall_\Gamma \mathcal{B})[\pi] \vdash \mathcal{A}} \qquad \frac{M \times \mathrm{Id}_\Gamma\ ;\ \overline{\mathcal{A}[\pi]} \vdash \mathcal{A}}{M\ ;\ \overline{\mathcal{A}} \vdash \forall_\Gamma A}\ (\textsc{Right}\ \forall)$$

Figure 1: Deduction rules on automata of Fig. 18 (§5.3), Fig. 20 (§6.3) and Fig. 24 (§7.2.4), where M, M' are composable, $\mathcal{N}, \overline{\mathcal{N}}$ are non-deterministic, and where the weakening functor $(-)[\pi]$ takes automata over Σ to automata over $\Sigma \times \Gamma$.

2.1. Compositionality and Categorical Semantics. The method of categorical semantics of proofs (see e.g. [LS86, AC98, Jac01, Mel09]) is to interpret *proofs* in a deduction system as *morphisms* in a category $\mathbb{C}$, such that $\mathbb{C}$ is equipped with some structure corresponding to the connectives and rules of the deduction system. For the moment, let us step back from acceptance games and consider run trees. Our task is thus to devise categories whose objects include all sets of the form $\mathcal{A}(T)$, for an automaton $\mathcal{A}$ and a tree T, and such that the proofs of a sequent $T\ ;\ \mathcal{A} \vdash \mathcal{B}$ can be interpreted as morphisms from $\mathcal{A}(T)$ to $\mathcal{B}(T)$.

The first requirement of categorical semantics is that the very notion of category already imposes interpretations to be *compositional*. Recall that the sets of morphisms of a (locally small) category $\mathbb{C}$ come with associative *composition* operations

$$(-)\circ(-) \quad : \quad \mathbb{C}[B,C]\times\mathbb{C}[A,B] \quad \longrightarrow \quad \mathbb{C}[A,C] \qquad\qquad \text{(for each } \mathbb{C}\text{-objects } A,B,C)$$

and with identity morphisms $\mathrm{id}_A \in \mathbb{C}[A,A]$ which are neutral for composition:

$$f\circ \mathrm{id}_A \quad = \quad f \quad = \quad \mathrm{id}_B \circ f \qquad \text{for every } f\in\mathbb{C}[A,B] \tag{10}$$

Composition and identities provide the interpretations respectively of the following instances of the usual *cut* and *axiom* rules:

$$(\textsc{Cut}_0)\ \frac{T\ ;\ \mathcal{A}\vdash\mathcal{B} \qquad T\ ;\ \mathcal{B}\vdash\mathcal{C}}{T\ ;\ \mathcal{A}\vdash\mathcal{C}} \qquad\qquad \frac{}{T\ ;\ \mathcal{A}\vdash\mathcal{A}}\ (\textsc{Axiom})$$

The identity laws (10) imply for instance that the three derivations below must be interpreted by the same morphism:

$$\frac{\dfrac{}{T\ ;\ \mathcal{A}\vdash\mathcal{A}} \qquad \dfrac{\mathscr{D}}{T\ ;\ \mathcal{A}\vdash\mathcal{B}}}{T\ ;\ \mathcal{A}\vdash\mathcal{B}} \qquad \frac{\mathscr{D}}{T\ ;\ \mathcal{A}\vdash\mathcal{B}} \qquad \frac{\dfrac{\mathscr{D}}{T\ ;\ \mathcal{A}\vdash\mathcal{B}} \qquad \dfrac{}{T\ ;\ \mathcal{B}\vdash\mathcal{B}}}{T\ ;\ \mathcal{A}\vdash\mathcal{B}} \tag{11}$$

2.2. Indexed Structure: Substitution and Quantification Rules. Our categories actually involve a slight generalization of the usual notion of acceptance (either with run trees or games) of automata. This generalization is induced by the axiomatization of quantification and substitution in categorical logic (see e.g. [Jac01, LS86]).

Let us briefly discuss the usual setting of first-order logic over a manysorted individual language. The categorical semantics of existential quantifications is given by an adjunction which is usually represented as

$$\frac{(\exists x)\varphi(x)\vdash\psi}{\varphi(x)\vdash\psi} \qquad (x \text{ not free in } \psi) \tag{12}$$

This adjunction induces a bijection between (the interpretations of) proofs of the sequents $\varphi(x)\vdash\psi$ and $(\exists x)\varphi(x)\vdash\psi$, that we informally denote

$$\varphi(x)\vdash\psi \qquad \simeq \qquad (\exists x)\varphi(x)\vdash\psi$$

Now, in general the variable x will occur free in φ. As a consequence, in order to properly formulate (12) one should be able to interpret sequents of the form $\varphi(x)\vdash\psi$ with free variables. More generally, the formulae φ and ψ should be allowed to contain free variables distinct from x.

The idea underlying the general method (but see e.g. [Jac01] for details), is to first devise a base category $\mathbb{B}$ of individuals, whose objects interpret products of sorts of the individual language, and whose maps from say $\iota_1\times\cdots\times\iota_m$ to $o_1\times\cdots\times o_n$ represent n-tuples $(t_1,\ldots,t_n)$

of terms t_i of sort o_i whose free variables are among $x_{\iota_1}, \ldots, x_{\iota_m}$ with x_{ι_j} of sort ι_j. Then, for each object $\iota = \iota_1 \times \cdots \times \iota_n$ of $\mathbb{B}$, one devises a category $\mathbb{E}_\iota$ whose objects represent formulae with free variables among $x_{\iota_1}, \ldots, x_{\iota_n}$, and whose morphisms interpret proofs. Furthermore, $\mathbb{B}$-morphisms

$$t = (t_1, \ldots, t_n) \quad : \quad \iota_1 \times \cdots \times \iota_m \quad \longrightarrow \quad o_1 \times \cdots \times o_n$$

induce *substitution functors*

$$t^\star \quad : \quad \mathbb{E}_{o_1 \times \cdots \times o_n} \quad \longrightarrow \quad \mathbb{E}_{\iota_1 \times \cdots \times \iota_m}$$

The functor $t^\star$ takes (the interpretation of) a formula φ with free variables among $y_{o_1}, \ldots, y_{o_n}$ to (the interpretation of) the formula $\varphi[t_1/y_{o_1}, \ldots, t_n/y_{o_n}]$ with free variables among $x_{\iota_1}, \ldots, x_{\iota_m}$. Its action on the morphisms of $\mathbb{E}_{o_1 \times \cdots \times o_n}$ allows us to interpret the *substitution rule*

$$\frac{\varphi \vdash \psi}{\varphi[t_1/y_{o_1}, \ldots, t_n/y_{o_n}] \vdash \psi[t_1/y_{o_1}, \ldots, t_n/y_{o_n}]}$$

In very good situations, the operation $(-)^\star$ is itself functorial. Among the morphisms of $\mathbb{B}$, one usually requires the existence of projections, say

$$\pi \quad : \quad o \times \iota \quad \longrightarrow \quad o$$

Projections induce substitution functors, called *weakening* functors

$$\pi^\star \quad : \quad \mathbb{E}_o \quad \longrightarrow \quad \mathbb{E}_{o \times \iota}$$

which simply allow to see formula $\psi(y_o)$ with free variable y_o as a formula $\psi(y_o, x_\iota)$ with free variables among y_o, x_ι (but with no actual occurrence of x_ι). Then the proper formulation of (12) is that existential quantification over x_ι is a functor

$$(\exists x_\iota)(-) \quad : \quad \mathbb{E}_{o \times \iota} \quad \longrightarrow \quad \mathbb{E}_o$$

which is left-adjoint to $\pi^\star$:

$$\frac{(\exists x_\iota)\varphi(x_\iota, y_o) \vdash \psi(y_o)}{\varphi(x_\iota, y_o) \vdash \pi^\star(\psi)(x_\iota, y_o)} \tag{13}$$

(where x_ι does not occur free in ψ since ψ is assumed to be (interpreted as) an object of $\mathbb{E}_o$, thus replacing the usual side condition). Universal quantifications are dually axiomatized as right adjoints to weakening functors. In both cases, the adjunctions are subject to additional conditions (called the *Beck-Chevalley* conditions) which ensure that they are preserved by substitution.

Returning to automata and infinite trees, we will take as base category the following category $\mathbf{T}$ of trees.

Definition 2.1 (The Base Category $\mathbf{T}$)**.** *The objects of* $\mathbf{T}$ *are alphabets, and its morphisms from* Σ *to* Γ*, denoted* $M, N, L, \ldots$*, are functions of the form*

$$\bigcup_{n \in \mathbb{N}} \left(\Sigma^{n+1} \times \mathfrak{D}^n\right) \quad \longrightarrow \quad \Gamma$$

A $\mathbf{T}$-morphism $M \in \mathbf{T}[\Sigma, \Gamma]$ thus takes for each $n \in \mathbb{N}$ a sequence of input letters $\overline{\mathtt{a}} = \mathtt{a}_0 \cdot \ldots \cdot \mathtt{a}_n \in \Sigma^{n+1}$ and a sequence of tree directions $p = d_1 \cdot \ldots \cdot d_n \in \mathfrak{D}^n$ to an output letter $M(\overline{\mathtt{a}}, p) \in \Gamma$. In particular, we have $\mathbf{T}[\mathbf{1}, \Sigma] \simeq (\mathfrak{D}^* \to \Sigma)$, so each Σ-labeled $\mathfrak{D}$-ary tree T corresponds to

a morphism $\dot{T} \in \mathbf{T}[\mathbf{1}, \Sigma]$. Moreover, $(\Sigma \to \Gamma)$-labeled trees $M : \mathfrak{D}^* \to (\Sigma \to \Gamma)$ induce $\mathbf{T}$-morphisms from Σ to Γ.[9] $\mathbf{T}$-morphisms are composed in the expected way (see §4.3 for details).

We will therefore not devise a single category $\mathbb{C}$, but a $\mathbf{T}$-indexed collection of categories $\mathbb{E}_\Sigma$, one for each alphabet Σ. Let us sketch the general idea with runs of non-deterministic automata. Given a non-deterministic automaton $\mathcal{A}$ over Γ and a morphism $M \in \mathbf{T}[\Sigma, \Gamma]$, a Σ-*run* of $\mathcal{A}$ on M is a tree

$$R \quad : \quad \mathfrak{D}^* \quad \longrightarrow \quad \Sigma \times Q_\mathcal{A}$$

such that $R(\epsilon) = (\mathtt{a}_0, q^i_\mathcal{A})$ for some $\mathtt{a}_0 \in \Sigma$, and which respects the transition function

$$\partial_\mathcal{A} \quad : \quad Q_\mathcal{A} \times \Gamma \quad \longrightarrow \quad \mathcal{P}(\mathfrak{D} \to Q_\mathcal{A})$$

supplied with input letters $\mathtt{b} \in \Gamma$ computed by M from tree positions $p = d_1 \cdot \ldots \cdot d_n$ and sequences of input letters $\overline{\mathtt{a}} = \mathtt{a}_0 \cdot \ldots \cdot \mathtt{a}_n$ where $\mathtt{a}_k$ is given by the Σ-component of $R(d_1 \cdot \ldots \cdot d_k) \in \Sigma \times Q_\mathcal{A}$. (So $\mathtt{a}_0$ is given by $R(\epsilon)$ and $\mathtt{a}_n$ is given by $R(p)$.) Explicitly, R is a Σ-run tree when for p and $\overline{\mathtt{a}}$ as above, if $R(p)$ is labeled with state $q \in Q_\mathcal{A}$, then there exists a $\mathfrak{D}$-tuple $(q_d)_{d \in \mathfrak{D}} \in \partial_\mathcal{A}(q, \mathtt{b})$ with $\mathtt{b} = M(\overline{\mathtt{a}}, p)$ and such that for all $d \in \mathfrak{D}$, $R(p \cdot d)$ is labeled with state q_d. Such a Σ-run R is *accepting* if the $Q_\mathcal{A}$-labeled tree

$$p \in \mathfrak{D}^* \quad \longmapsto \quad \pi(R(p)) \in Q_\mathcal{A}$$

is accepting in the usual sense (where $\pi : \Sigma \times Q_\mathcal{A} \to Q_\mathcal{A}$ is the second projection), that is if all its infinite paths belong to $\Omega_\mathcal{A}$. We let $\Sigma \vdash \mathcal{A}(M)$ be the set of accepting Σ-run trees of $\mathcal{A}$ on M, and simply write $\mathcal{A}(M)$ for $\Sigma \vdash \mathcal{A}(M)$ when Σ is clear from the context.

Roughly speaking, for each Σ, the objects of the category $\mathbb{E}_\Sigma$ will include all sets of the form $\Sigma \vdash \mathcal{A}(M)$. Moreover, given $L \in \mathbf{T}[\Delta, \Sigma]$, the substitution functor

$$L^\star \quad : \quad \mathbb{E}_\Sigma \quad \longrightarrow \quad \mathbb{E}_\Delta$$

will take an $\mathbb{E}_\Sigma$-object $\Sigma \vdash \mathcal{A}(M)$ to the $\mathbb{E}_\Delta$-object $\Delta \vdash \mathcal{A}(M \circ L)$, where the $\mathbf{T}$-map $L \circ M \in \mathbf{T}[\Delta, \Gamma]$ is the $\mathbf{T}$-composition of L and M (assuming $M \in \mathbf{T}[\Sigma, \Gamma]$ as above).

This will induce sequents generalizing (4). For instance, given $M \in \mathbf{T}[\Sigma, \Gamma]$, we have sequents of the form

$$M \ ; \ \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B} \tag{14}$$

where $\mathcal{A}_1, \ldots, \mathcal{A}_n$ and $\mathcal{B}$ are automata over Γ. Such sequents are to be thought about as our version of "*open* sequents" or "sequents with free variables" (here of sort Σ), with the usual implicit prenex universal quantification, and are to be interpreted as morphisms in the category $\mathbb{E}_\Sigma$ (the *fibre* over Σ). Substitution functors such as $L^\star : \mathbb{E}_\Sigma \to \mathbb{E}_\Delta$ above will act in the deduction system via a substitution rule

$$(\textsc{Subst}) \quad \frac{M \ ; \ \mathcal{A}_1, \ldots \mathcal{A}_n \vdash \mathcal{B}}{M \circ L \ ; \ \mathcal{A}_1, \ldots \mathcal{A}_n \vdash \mathcal{B}} \qquad (\text{where } M \in \mathbf{T}[\Sigma, \Gamma] \text{ and } L \in \mathbf{T}[\Delta, \Sigma]) \tag{15}$$

Let us briefly sketch the most important instances of this construction.

(a) Consider a $\mathbf{T}$-map $\dot{T} : \mathbf{T}[\mathbf{1}, \Sigma]$ representing a tree $T : \mathfrak{D}^* \to \Sigma$. Then the accepting runs of $\mathcal{A}$ on T are in bijection with the accepting $\mathbf{1}$-run trees of $\mathcal{A}$ on $\dot{T}$:

$$(\mathbf{1} \vdash \mathcal{A}(\dot{T})) \quad \simeq \quad \mathcal{A}(T)$$

[9] The morphisms from Σ to Γ of the base category of [Rib15] are restricted to $(\Sigma \to \Gamma)$-labeled trees.

Sequents of the form (14) thus indeed generalize sequents of the form

$$T \; ; \; \mathcal{A}_1 \, , \, \ldots \, , \, \mathcal{A}_n \vdash \mathcal{B}$$

with $T : \mathfrak{D}^* \to \Sigma$ (as depicted in (4)), which are to be interpreted in the category $\mathbb{E}_{\mathbf{1}}$ (the fibre over $\mathbf{1}$), and are to be thought about as representing *closed* statements.

(b) Given a non-deterministic automaton $\mathcal{A}$ over Σ, we write $\Sigma \vdash \mathcal{A}$ (or even just $\mathcal{A}$ when no ambiguity arises) for $\Sigma \vdash \mathcal{A}(\mathrm{Id}_\Sigma)$ where the $\mathbf{T}$-identity $\mathrm{Id}_\Sigma \in \mathbf{T}[\Sigma, \Sigma]$ is given by

$$\mathrm{Id}_\Sigma(\overline{\mathtt{a}} \cdot \mathtt{a}, p) \quad := \quad \mathtt{a}$$

Consider now another automaton $\mathcal{B}$, also over Σ. Then we write

$$\Sigma \; ; \; \mathcal{A} \vdash \mathcal{B} \tag{16}$$

(or even $\mathcal{A} \vdash \mathcal{B}$) for the sequent $\mathrm{Id}_\Sigma \; ; \; \mathcal{A} \vdash \mathcal{B}$. The *provability interpretation* of (16) will be that if (16) is provable, then $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})$. The *computational interpretation* of (16) will consist in a form of uniform simulation of $\mathcal{A}$ by $\mathcal{B}$ (generalizing the notion used with the *guidable automata* of [CL08]). Moreover, given a Σ-labeled tree T seen as a morphism $\dot{T} \in \mathbf{T}[\mathbf{1}, \Sigma]$, the interpretation of the substitution rule

$$\frac{\Sigma \; ; \; \mathcal{A} \vdash \mathcal{B}}{\dot{T} \; ; \; \mathcal{A} \vdash \mathcal{B}}$$

will take a morphism $\sigma \in \mathbb{E}_\Sigma[\mathcal{A}, \mathcal{B}]$ to a function $\dot{T}^\star(\sigma) : \mathcal{A}(T) \to \mathcal{B}(T)$.

(c) Any ordinary function $\mathtt{f} : \Sigma \to \Gamma$ induces a morphism $[\mathtt{f}] \in \mathbf{T}[\Sigma, \Gamma]$ defined as

$$[\mathtt{f}] \quad : \quad (\overline{\mathtt{a}} \cdot \mathtt{a}, p) \quad \longmapsto \quad \mathtt{f}(\mathtt{a})$$

The action of the substitution functor $[\mathtt{f}]^\star : \mathbb{E}_\Gamma \to \mathbb{E}_\Sigma$ on $\mathbb{E}_\Gamma$-objects of the form $\Gamma \vdash \mathcal{A}$ can be internalized in automata. We indeed have

$$[\mathtt{f}]^\star(\Gamma \vdash \mathcal{A}) \quad = \quad \Sigma \vdash \mathcal{A}([\mathtt{f}]) \quad = \quad \Sigma \vdash \mathcal{A}[\mathtt{f}]$$

where the automaton $\mathcal{A}[\mathtt{f}]$ over Σ is defined as $\mathcal{A}$ but with transition function:

$$\begin{array}{rcccl} \partial_{\mathcal{A}[\mathtt{f}]} & : & Q_\mathcal{A} \times \Gamma & \longrightarrow & \mathcal{P}(\mathfrak{D} \to Q_\mathcal{A}) \\ & & (q, \mathtt{b}) & \longmapsto & \partial_\mathcal{A}(q, \mathtt{f}(\mathtt{b})) \end{array}$$

In particular:

(i) $\mathbf{T}$-maps from $\Sigma \times \Gamma$ to Σ indeed include projections $[\pi] : \mathfrak{D}^* \to (\Sigma \times \Gamma \to \Sigma)$ induced by $\mathbf{Set}$-projections $\pi : \Sigma \times \Gamma \to \Sigma$.

(ii) Consider automata $\mathcal{A}_1, \ldots, \mathcal{A}_n$ and $\mathcal{B}$, with $\mathcal{A}_i$ over Σ_i and $\mathcal{B}$ over Γ. Consider furthermore $\mathbf{T}$-morphisms $M_i \in \mathbf{T}[\Delta, \Sigma_i]$ and $L \in \mathbf{T}[\Delta, \Gamma]$. Then we write

$$\Delta \; ; \; \mathcal{A}_1(M_1) \, , \, \ldots \, , \, \mathcal{A}_n(M_n) \vdash \mathcal{B}(L)$$

for the sequent

$$\langle M_1, \ldots, M_n, L \rangle \; ; \; \mathcal{A}_1[\pi_1] \, , \, \ldots \, , \, \mathcal{A}_n[\pi_n] \vdash \mathcal{B}[\pi]$$

where

$$\langle M_1, \ldots, M_n, L\rangle \;\in\; \mathbf{T}[\Delta,\, \Sigma_1 \times \cdots \times \Sigma_n \times \Gamma]$$

is the $\mathbf{T}$-tupling of $M_1, \ldots, M_n, L$ (see Cor. 4.6) and where the π_i's and π are suitable projections:

$$\begin{array}{lcccl} \pi_i & : & \Sigma_1 \times \cdots \times \Sigma_n \times \Gamma & \longrightarrow & \Sigma_i \\ \pi & : & \Sigma_1 \times \cdots \times \Sigma_n \times \Gamma & \longrightarrow & \Gamma \end{array}$$

Unless otherwise stated, all the sequents seen up to now must from now on be thought about as being of the more general form (16), that is a with a $\mathbf{T}$-map M (of appropriate type) instead of the labeled tree T.

2.3. Toward a Semantics for Implications. The *provability interpretation* of sequents tells us that in sequents of the form

$$M \;;\; \mathcal{A} \vdash \mathcal{B} \tag{17}$$

the symbol $\vdash$ is a form of implication. One of the main contributions of this work is that this implication can be internalized in automata. This will lead us outside of non-deterministic automata (see §3), but for the moment let us sketch some salient consequences this imposes to the interpretation of the symbol $\vdash$ in sequents of the form (17).

Assume that proofs of our deduction system are interpreted in categories $\mathbb{E}_{(-)}$ indexed over $\mathbf{T}$. Then, internalizing $\vdash$ in automata will imply that given automata $\mathcal{A}$ and $\mathcal{B}$ over Σ there is an automaton $(\mathcal{A} \multimap \mathcal{B})$ over Σ such that for each tree $T : \mathfrak{D}^* \to \Sigma$ there is a bijection

$$\mathbb{E}_{\mathbf{1}}[\mathcal{A}(\dot{T}),\, \mathcal{B}(\dot{T})] \quad \simeq \quad \mathbf{1} \vdash (\mathcal{A} \multimap \mathcal{B})(\dot{T})$$

that we informally write as

$$\dot{T} \;;\; \mathcal{A} \vdash \mathcal{B} \qquad \simeq \qquad \mathbf{1} \vdash (\mathcal{A} \multimap \mathcal{B})(\dot{T})$$

In other words, morphisms in the interpretation of $\dot{T} \;;\; \mathcal{A} \vdash \mathcal{B}$ will correspond to the runs of an automaton $(\mathcal{A} \multimap \mathcal{B})$ on T. This could suggest to interpret $\dot{T} \;;\; \mathcal{A} \vdash \mathcal{B}$ as the runs of an automaton of the form $\sim\!\mathcal{A} \vee \mathcal{B}$ over T, where $\sim\!\mathcal{A}$ is the complement of $\mathcal{A}$ (in the sense of §1.2) and $(-) \vee (-)$ is a disjunction on automata. Let us rule out this possibility, at least for the natural implementation of $(-) \vee (-)$ with an *additive* disjunction $(-) \oplus (-)$. Given automata $\mathcal{A}_1$ and $\mathcal{A}_2$, both over Σ and with $\mathcal{A}_i = (Q_{\mathcal{A}_i}, q^{\imath}_{\mathcal{A}_i}, \partial_{\mathcal{A}_i}, \Omega_{\mathcal{A}_i})$, the non-deterministic automaton $\mathcal{A}_1 \oplus \mathcal{A}_2$ over Σ is

$$\mathcal{A}_1 \oplus \mathcal{A}_2 \quad := \quad (Q_{\mathcal{A}_1} + Q_{\mathcal{A}_2} + \mathbf{1}\,,\, \bullet\,,\, \partial_{\mathcal{A}_1 \oplus \mathcal{A}_2}\,,\, \Omega_{\mathcal{A}_1 \oplus \mathcal{A}_2})$$

where, via the embedding of $Q^{\mathfrak{D}}_{\mathcal{A}_1} + Q^{\mathfrak{D}}_{\mathcal{A}_2}$ into $(Q_{\mathcal{A}_1} + Q_{\mathcal{A}_2})^{\mathfrak{D}}$, we let

$$\partial_{\mathcal{A}_1 \oplus \mathcal{A}_2}(q, \mathsf{a}) \quad := \quad \begin{cases} \partial_{\mathcal{A}_1}(q^{\imath}_{\mathcal{A}_1}, \mathsf{a}) + \partial_{\mathcal{A}_2}(q^{\imath}_{\mathcal{A}_2}, \mathsf{a}) & \text{if } q = \bullet \in \mathbf{1} \\ \partial_{\mathcal{A}_i}(q, \mathsf{a}) & \text{if } q \in Q_{\mathcal{A}_i} \end{cases}$$

and where $\bullet, q_1, q_2, \ldots \in \Omega_{\mathcal{A}_1 \oplus \mathcal{A}_2}$ iff either $q^{\imath}_{\mathcal{A}_1}, q_1, q_2, \ldots \in \Omega_{\mathcal{A}_1}$ or $q^{\imath}_{\mathcal{A}_2}, q_1, q_2, \ldots \in \Omega_{\mathcal{A}_2}$.

Note that in **Set**, for every $M : \mathfrak{D}^* \to (\Gamma \to \Sigma)$ we have

$$(\mathcal{A}_1 \oplus \mathcal{A}_2)(M) \quad \simeq \quad \mathcal{A}_1(M) + \mathcal{A}_2(M)$$

so in particular

$$\mathcal{L}(\mathcal{A}_1 \oplus \mathcal{A}_2) \quad = \quad \mathcal{L}(\mathcal{A}_1) \cup \mathcal{L}(\mathcal{A}_2)$$

Assume now that we take for $\mathbb{E}_{\mathbf{1}}[\mathcal{A}(\dot{T}),\, \mathcal{B}(\dot{T})]$ the set of runs of $(\sim\!\mathcal{A} \oplus \mathcal{B})$ on T, that is the disjoint union $\sim\!\mathcal{A}(T) + \mathcal{B}(T)$. Then one faces the following difficulties.

- We have to devise identity morphisms, say

$$\mathrm{id}_{\mathcal{A}(\dot{T})} \quad \in \quad \sim\mathcal{A}(T) + \mathcal{A}(T)$$

  One may take for $\mathrm{id}_{\mathcal{A}(\dot{T})}$ either an accepting run of $\mathcal{A}$ on T or an accepting run of $\sim\mathcal{A}$ on T. But this raises two problems. First, it may be undecidable whether a possibly non-recursive tree is accepted or rejected by a given automaton. So this precludes any *general and effective* computational interpretation of the deduction system. Second, even if we restrict to trees T for which acceptance is known to be decidable (e.g. trees generated by *higher-order recursion schemes* [Ong06]), there seem to be no *canonical choice* of an actual accepting run $\mathrm{id}_{\mathcal{A}(\dot{T})} \in \sim\mathcal{A}(T) + \mathcal{A}(T)$.

- It is not clear how to define composition, say

$$(-) \circ (-) \quad : \quad (\sim\mathcal{B}(T) + \mathcal{C}(T)) \times (\sim\mathcal{A}(T) + \mathcal{B}(T)) \quad \longrightarrow \quad \sim\mathcal{A}(T) + \mathcal{C}(T)$$

  Given run trees, say

$$R_{\mathcal{C}(T)} \ \in \ \mathcal{C}(T) \ \subseteq \ \sim\mathcal{B}(T) + \mathcal{C}(T) \qquad \text{and} \qquad R_{\sim\mathcal{A}(T)} \ \in \ \sim\mathcal{A}(T) \ \subseteq \ \sim\mathcal{A}(T) + \mathcal{B}(T)$$

  there seems to be no obvious choice for $R_{\mathcal{C}(T)} \circ R_{\sim\mathcal{A}(T)} \in \sim\mathcal{A}(T) + \mathcal{C}(T)$. Both

$$R_{\mathcal{C}(T)} \circ R_{\sim\mathcal{A}(T)} \quad := \quad R_{\mathcal{C}(T)} \qquad \text{and} \qquad R_{\mathcal{C}(T)} \circ R_{\sim\mathcal{A}(T)} \quad := \quad R_{\sim\mathcal{A}(T)}$$

  may seem reasonable. But each of them breaks one of the equalities between the interpretations of the derivations depicted in (11).

The methodology of linear logic may suggest here to devise a linear implication of the form

$$\mathcal{A} \multimap \mathcal{B} \quad := \quad \mathcal{A}^{\perp} \parr \mathcal{B}$$

where $\parr$ is a dual of the direct product $\otimes$ (see §1.3 and §2.4 below), relying on a Cartesian product of states and evaluating its arguments in parallel, with acceptance given by a disjunction. However, in contrast with ω-word automata [PR18], such a connective does not seem to exist on tree automata. The reason is that the universal quantification on paths (in the definition of acceptance) does not commute with disjunction.

2.4. The (Synchronous) Direct Product of (Non-Deterministic) Automata. The solution to obtain a suitable *closed* structure will be provided by uniform automata, to be defined in §3. On the other hand, part of the program announced up to now was already completed in [Rib15]. In that work, using the notions of *substituted acceptance games* and of *synchronous linear arrow games*, we obtained categories of (usual) alternating automata fulfilling the requirements of §2.1 and §2.2. Although [Rib15] does not explicitly mention any deduction system, it does devise categorical structures allowing for parts of the linear logic based approach mentioned in §1.3.

We survey here the relevant connections between [Rib15] and IMELL, as they underlie the role of linear logic in this paper. Returning to the general case of sequents of the form

$$M \ ; \ \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B} \tag{18}$$

the *provability interpretation* tells us that the left commas correspond to a form of conjunction. A conjunction on non-deterministic automata can be implemented with a direct (synchronous) product. The *direct product* $\mathcal{A}_1 \otimes \mathcal{A}_2$ of the non-deterministic automata $\mathcal{A}_i = (Q_{\mathcal{A}_i}, q^\imath_{\mathcal{A}_i}, \partial_{\mathcal{A}_i}, \Omega_{\mathcal{A}_i})$, both over Σ, is the non-deterministic automaton over Σ

$$\mathcal{A}_1 \otimes \mathcal{A}_2 \quad := \quad \left(Q_{\mathcal{A}_1} \times Q_{\mathcal{A}_2}\,,\, (q^\imath_{\mathcal{A}_1}, q^\imath_{\mathcal{A}_2})\,,\, \partial_{\mathcal{A}_1 \otimes \mathcal{A}_2}\,,\, \Omega_{\mathcal{A}_1 \otimes \mathcal{A}_2}\right)$$

with

$$\partial_{\mathcal{A}_1 \otimes \mathcal{A}_2}((q_1, q_2), \mathtt{a}) \quad := \quad \{\langle \mathtt{g}_1, \mathtt{g}_2 \rangle : \mathfrak{D} \to Q_{\mathcal{A}_1} \times Q_{\mathcal{A}_2} \mid \mathtt{g}_i \in \partial_{\mathcal{A}_i}(q_i, \mathtt{a}) \text{ for } i = 1, 2\}$$

and where $\Omega_{\mathcal{A}_1 \otimes \mathcal{A}_2}$ is $\Omega_{\mathcal{A}_1} \times \Omega_{\mathcal{A}_2}$ modulo $(Q_{\mathcal{A}_1} \times Q_{\mathcal{A}_2})^\omega \simeq Q^\omega_{\mathcal{A}_1} \times Q^\omega_{\mathcal{A}_2}$. For every tree T, the (accepting) runs of $\mathcal{A}_1 \otimes \mathcal{A}_2$ on T are exactly[10] the pairs $\langle R_1, R_2 \rangle : \mathfrak{D}^* \to Q_{\mathcal{A}_1} \times Q_{\mathcal{A}_2}$ of (accepting) runs of $\mathcal{A}_1$ and $\mathcal{A}_2$ over T. We therefore have, in the category **Set**

$$(\mathcal{A}_1 \otimes \mathcal{A}_2)(T) \quad \simeq \quad \mathcal{A}_1(T) \times \mathcal{A}_2(T) \tag{19}$$

from which we immediately get

$$\mathcal{L}(\mathcal{A}_1 \otimes \mathcal{A}_2) \quad = \quad \mathcal{L}(\mathcal{A}_1) \cap \mathcal{L}(\mathcal{A}_2)$$

For similar reasons, the direct product $(-) \otimes (-)$ on non-deterministic automata is also Cartesian in the games of [Rib15] provided one restricts to *total* automata.[11] This Cartesian structure on total non-deterministic automata implies that we can equip them with the deduction rules of a Cartesian product, such as the following (where $\mathbf{I}$ is a unit automaton similar to that of Ex. 3.2.(i)):

$$\begin{array}{llll} (\textsc{Left}\ \otimes) & \dfrac{M\ ;\ \overline{\mathcal{A}},\, \mathcal{A},\, \mathcal{B},\, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\, \mathcal{A} \otimes \mathcal{B},\, \overline{\mathcal{B}} \vdash \mathcal{C}} & \dfrac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\ ;\ \overline{\mathcal{B}} \vdash \mathcal{B}}{M\ ;\ \overline{\mathcal{A}},\, \overline{\mathcal{B}} \vdash \mathcal{A} \otimes \mathcal{B}} & (\textsc{Right}\ \otimes) \\ (\textsc{Left}\ \mathbf{I}) & \dfrac{M\ ;\ \overline{\mathcal{A}},\, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\, \mathbf{I},\, \overline{\mathcal{B}} \vdash \mathcal{C}} & \dfrac{}{M\ ;\ \vdash \mathbf{I}} & (\textsc{Right}\ \mathbf{I}) \end{array} \tag{20}$$

together with the structural *exchange rule*:

$$(\textsc{Exchange})\ \ \frac{M\ ;\ \overline{\mathcal{A}},\, \mathcal{A},\, \mathcal{B},\, \overline{\mathcal{C}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\, \mathcal{B},\, \mathcal{A},\, \overline{\mathcal{C}} \vdash \mathcal{C}} \tag{21}$$

as well as the structural *weakening* and *contraction* rules:

$$(\textsc{Weak})\ \ \frac{M\ ;\ \overline{\mathcal{A}},\, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\, \mathcal{A},\, \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{A}},\, \mathcal{A},\, \mathcal{A},\, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\, \mathcal{A},\, \overline{\mathcal{B}} \vdash \mathcal{C}}\ (\textsc{Contr}) \tag{22}$$

and the following general (multiplicative) cut rule:

$$(\textsc{Cut})\ \ \frac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\ ;\ \overline{\mathcal{B}},\, \mathcal{A},\, \overline{\mathcal{C}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{B}},\, \overline{\mathcal{A}},\, \overline{\mathcal{C}} \vdash \mathcal{C}} \tag{23}$$

To summarize, with total non-deterministic automata, the left commas in sequents of the form (18) can be internalized as a product $(\otimes, \mathbf{I})$, whose deduction rules are induced by its structure in the computational interpretation.

[10] Because universal quantifications commute over conjunctions!

[11] We say that a non-deterministic automaton $\mathcal{A}$ is *total* if the empty set is not in the range of its transition function.

2.5. Alternating Automata and Linear Logic. With respect to the context of this paper, the basic insight of linear logic [Gir87], is that having an explicit control on the weakening and contraction structural rules depicted in (22) gives rise to a decomposition of the usual intuitionistic connectives $\wedge, \rightarrow$ into more refined connectives (usually denoted $\otimes, \&, !, \multimap$), which in a lot of cases allow, thanks to the Curry-Howard correspondence, refined constructions of models of programming languages based on (typed) λ-calculi (see e.g. [AC98]).

In the case of conjunction, this can be phrased as follows. First, when suppressing the structural rules (Weak) and (Contr), the rules displayed in (20) and (21) do not specify anymore a Cartesian structure for the product $(\otimes, \mathbf{I})$, but merely a symmetric monoidal structure (see e.g. [Mel09] for definitions). This implies that in contrast with intuitionistic sequents, the left commas in linear sequents, which have the same structure as $(\otimes, \mathbf{I})$, do not anymore behave as a Cartesian product. Moreover, $(\otimes, \mathbf{I})$ is not anymore equivalent to the *additive* conjunction (usually denoted $\&$, with unit $\top$), which, as a logical connective, would be defined in linear sequents by rules of the form[12]:

$$\frac{A_1\,,\,\ldots\,,\,A_n \vdash B_1 \qquad A_1\,,\,\ldots\,,\,A_n \vdash B_2}{A_1\,,\,\ldots\,,\,A_n \vdash B_1 \,\&\, B_2} \qquad \frac{}{A_1\,,\,\ldots\,,\,A_n \vdash \top}$$

$$\frac{A_1\,,\,\ldots\,,\,A_i\,,\,\ldots\,,\,A_n \vdash B}{A_1\,,\,\ldots\,,\,A_i \,\&\, C\,,\,\ldots\,,\,A_n \vdash B} \qquad \frac{A_1\,,\,\ldots\,,\,A_i\,,\,\ldots\,,\,A_n \vdash B}{A_1\,,\,\ldots\,,\,C \,\&\, A_i\,,\,\ldots\,,\,A_n \vdash B}$$

Second, the structural rules (Weak) and (Contr) are restored but for a specific *exponential modality* $!(-)$:

$$\frac{A_1\,,\,\ldots\,,\,\ldots\,,\,A_n \vdash B}{A_1\,,\,\ldots\,,\,!A_i\,,\,\ldots\,,\,A_n \vdash B} \qquad \frac{A_1\,,\,\ldots\,,\,!A_i\,,\,!A_i\,,\,\ldots A_n \vdash B}{A_1\,,\,\ldots\,,\,!A_i\,,\,\ldots\,,\,A_n \vdash B} \tag{24}$$

The modality $!(-)$ is itself subject to specific introduction rules, called *dereliction* and *promotion*:

$$\frac{A_1\,,\,\ldots\,,\,A_i\,,\,\ldots\,,\,A_n \vdash B}{A_1\,,\,\ldots\,,\,!A_i\,,\,\ldots\,,\,A_n \vdash B} \qquad \frac{!A_1\,,\,\ldots\,,\,!A_n \vdash B}{!A_1\,,\,\ldots\,,\,!A_n \vdash !B} \tag{25}$$

Then (but see also [Gir87, AC98, Mel09] for details), the categorical interpretation of proofs gives an isomorphism

$$!A \otimes !B \quad \simeq \quad !(A \,\&\, B) \tag{26}$$

which implies that an intuitionistic sequent

$$A_1\,,\,\ldots\,,\,A_n \vdash B$$

where the left commas behave as a Cartesian product, corresponds to the linear sequent

$$!A_1\,,\,\ldots\,,\,!A_n \vdash B$$

where the left commas behave as a symmetric monoidal product $(-) \otimes (-)$.

The pertinence of intuitionistic linear logic in our context comes from the fact that the product $(-)\otimes(-)$ defined in §2.4 on non-deterministic automata extends to (total[13]) alternating automata, but induces a symmetric monoidal product which is not Cartesian.

[12] We do not consider in this paper the usual additive conjunction on alternating automata, which would provide an implementation of $\&$, because its categorical properties would require a slight extension of our setting.

[13] *Total* alternating automata were called *complete* in [Rib15].

Given alternating automata $\mathcal{A}$ and $\mathcal{B}$ over Σ, the automaton $\mathcal{A} \otimes \mathcal{B}$ over Σ has state set $Q_{\mathcal{A}} \times Q_{\mathcal{B}}$, and evaluates $\mathcal{A}$ and $\mathcal{B}$ along common paths $p \in \mathfrak{D}^*$ (see [Rib15] for details). Now, recall that with alternating automata, O can choose states in addition to tree directions. Hence, given a P-strategy on $(\mathcal{A} \otimes \mathcal{B})(T)$ (for $T : \mathfrak{D}^* \to \Sigma$), and given a branch of this strategy following a given path $p \in \mathfrak{D}^*$, it is possible for P to make different choices according to previous O-moves. In particular, some choice of P in component $\mathcal{A}$ may depend on previous O-moves in $\mathcal{B}$. (Note that this was not possible with non-deterministic automata, since $p \in \mathfrak{D}^*$ determines uniquely the previous O-moves.) So a P-strategy on $(\mathcal{A} \otimes \mathcal{B})(T)$ may not uniquely determine a pair of strategies in $\mathcal{A}(T) \times \mathcal{B}(T)$.

On the other hand, in any model of intuitionistic linear logic, the isomorphism (26) implies that every object of the form $!A$ is a commutative comonoid w.r.t. $(\otimes, \mathbf{I})$ (see e.g. [Mel09]), which essentially means that $(\otimes, \mathbf{I})$ has a Cartesian structure w.r.t. objects of the form $!A$. This indicates that non-deterministic automata behave as objects of the form $!A$, and it turns out that to some extent, the powerset construction translating an alternating automaton to an equivalent non-deterministic one (the *Simulation Theorem* [MS87, EJ91, MS95]), corresponds to an $!(-)$-modality of intuitionistic linear logic. In particular, all the $!(-)$-rules (24) and (25) can be interpreted in our categories[14]. (But unfortunately, this interpretation is not compatible with usual cut-elimination, because the operation $!(-)$ fails to be a functor.)

3. Uniform Automata and Zig-Zag Strategies

In view of §2.3, it seems that the categorical structure required for a Curry-Howard approach should involve some machinery not given by usual connectives on automata. In [Rib15], we proposed categories of generalized acceptance games based on the technology of *game semantics*, and more precisely on *simple games* (see e.g. [Abr97, Hyl97]), which stem from Berry & Curien's *sequential data structures* (see e.g. [AC98, Chap. 14], but also [Mel05]). The model of [Rib15] uses usual alternating automata, which seem unfortunately not to induce categories equipped with a monoidal *closed* structure while providing a computational interpretation of proofs in the sense of §1.4.

We present here the notion of automata (called *uniform* automata) on which this paper relies (§3.1 and §3.2), as well as the adaption of the substituted acceptance games of [Rib15] to this context (§3.3). Uniform automata are motivated by the extension of usual alternating automata with a monoidal *closed* structure. Working with uniform automata instead of usual automata allows, w.r.t. [Rib15], for a considerable simplification of the underlying technology of game semantics. We rely on a very simple category **DZ** of (total) *zig-zag* games presented in §3.4 (**DZ** stands for *Dialectica*-like *Zig-zag* games, see Rem. 3.11), on top of which we build the counterpart for uniform automata of substituted acceptance games and synchronous arrows games (§3.5).

The proof that **DZ** is a category is deferred to App. A, which also discusses the connections between our approach and usual simple games.

3.1. Uniform Automata. In order to obtain the required categorical properties of a monoidal closed structure, we devise a "uniform" variant of usual alternating automata, whose transitions are given by explicit arbitrary non-empty finite sets of P and O-moves. The corresponding monoidal closed structure is presented in §5.

[14] (Weak) actually holds (in a non-canonical way) for total alternating automata (*i.e.* the ! is not strictly necessary in the conclusion).

Definition 3.1 (Uniform Tree Automata)**.** *A* uniform tree automaton $\mathcal{A}$ *over* Σ *(notation* $\mathcal{A} : \Sigma$*) has the form*

$$\mathcal{A} \quad = \quad (Q_{\mathcal{A}}\,,\, q^{\imath}_{\mathcal{A}}\,,\, U\,,\, X\,,\, \partial_{\mathcal{A}}\,,\, \Omega_{\mathcal{A}}) \tag{27}$$

where $Q_{\mathcal{A}}$ *is the finite set of* states, $q^{\imath}_{\mathcal{A}} \in Q_{\mathcal{A}}$ *is the* initial state, U *and* X *are finite non-empty sets of resp.* P *and* O*-moves, the* acceptance condition $\Omega_{\mathcal{A}}$ *is an* ω*-regular subset of* $Q^{\omega}_{\mathcal{A}}$*, and the* transition function $\partial_{\mathcal{A}}$ *has the form*

$$\partial_{\mathcal{A}} \quad : \quad Q_{\mathcal{A}} \times \Sigma \quad \longrightarrow \quad U \times X \quad \longrightarrow \quad (\mathfrak{D} \longrightarrow Q_{\mathcal{A}}) \tag{28}$$

Following the usual terminology, an automaton $\mathcal{A}$ *as in (27) is* non-deterministic *if* $X \simeq \mathbf{1}$*,* universal *if* $U \simeq \mathbf{1}$*, and* deterministic *if* $U \simeq X \simeq \mathbf{1}$*.*

Example 3.2. *(i) The* unit automaton $\mathbf{I}_{\Sigma} : \Sigma$ *is the unique uniform deterministic automaton over* Σ *with state set* $\mathbf{1}$ *(with* $\bullet$ *initial) and acceptance condition* $\mathbf{1}^{\omega}$*. Explicitly,*

$$\mathbf{I}_{\Sigma} \quad := \quad (\mathbf{1}, \bullet, \mathbf{1}, \mathbf{1}, \partial_{\mathbf{1}}, \mathbf{1}^{\omega})$$

where $\partial_{\mathbf{1}}$ *is the unique function*

$$\partial_{\mathbf{1}} \quad : \quad \mathbf{1} \times \Sigma \quad \longrightarrow \quad \mathbf{1} \times \mathbf{1} \quad \longrightarrow \quad (\mathfrak{D} \longrightarrow \mathbf{1})$$

We write $\mathbf{I}$ *for* $\mathbf{I}_{\Sigma}$ *when* Σ *is clear from the context.*

(ii) Each alternating automaton $\mathcal{A}$ *can be translated to a uniform automaton* $\widehat{\mathcal{A}}$*. The automaton* $\widehat{\mathcal{A}}$ *simulates* $\mathcal{A}$ *as long as* P *and* O *respect the transition function of* $\mathcal{A}$*, and switches to an accepting (resp. rejecting) state as soon as* O *(resp.* P*) plays a move not allowed by* $\mathcal{A}$*. Assuming*

$$\partial_{\mathcal{A}} \quad : \quad Q_{\mathcal{A}} \times \Sigma \quad \longrightarrow \quad \mathcal{P}(\mathcal{P}(Q_{\mathcal{A}} \times \mathfrak{D}))$$

we let $\widehat{\mathcal{A}}$ *be the uniform automaton*

$$(\widehat{\mathcal{A}} : \Sigma) \quad := \quad (Q_{\mathcal{A}} + \mathbb{B}\,,\, q^{\imath}_{\mathcal{A}}\,,\, \mathcal{P}(Q_{\mathcal{A}} \times \mathfrak{D})\,,\, Q_{\mathcal{A}}\,,\, \partial_{\widehat{\mathcal{A}}}\,,\, \Omega_{\widehat{\mathcal{A}}})$$

where $\mathbb{B} := \{\mathbb{t}, \mathbb{f}\}$*, with transitions given by* $\partial_{\widehat{\mathcal{A}}}(\mathbb{b}, \mathsf{a}, -, -, -) := \mathbb{b}$ *if* $\mathbb{b} \in \mathbb{B}$ *and for* $q \in Q_{\mathcal{A}}$*:*

$$\partial_{\widehat{\mathcal{A}}}(q, \mathsf{a}, \gamma, q', d) := \begin{cases} q' & \text{if } \gamma \in \partial_{\mathcal{A}}(q, \mathsf{a}) \text{ and } (q', d) \in \gamma \\ \mathbb{t} & \text{if } \gamma \in \partial_{\mathcal{A}}(q, \mathsf{a}) \text{ and } (q', d) \notin \gamma \\ \mathbb{f} & \text{if } \gamma \notin \partial_{\mathcal{A}}(q, \mathsf{a}) \end{cases}$$

and with $\Omega_{\widehat{\mathcal{A}}} \;:=\; \Omega_{\mathcal{A}} + Q^{*}_{\mathcal{A}}.\mathbb{t}^{\omega}$*.*

3.2. Full Positive Games and Acceptance for Uniform Automata. The shape (28) of the transition functions of uniform automata allows for their acceptance games to be defined without imposing legality conditions on plays. This leads to a slightly simpler setting than for usual automata.

Definition 3.3 (Full Positive Games)**.**

- *A* full positive game *has the form* $A = (U, X)$ *where* U *and* X *are sets of resp.* P *and* O-moves*. We say that* $A = (U, X)$ *is* total *if* U *and* X *are both non-empty.*

	$\mathcal{A}(T)$
	$(\epsilon, q^{\iota}_{\mathcal{A}})$
	$\vdots$
	$(p, q_{\mathcal{A}})$
P	u
O	(x, d)
	$(p.d, q'_{\mathcal{A}})$
P	u'
O	(x', d')
	$(p.d.d', q''_{\mathcal{A}})$
	$\vdots$

Figure 2: Acceptance game for uniform automata

- *A* full positive game with winning condition *is a full positive game* $A = (U, X)$ *together with a* winning condition $\mathcal{W}_A \subseteq (U \cdot X)^{\omega}$.

A typical (infinite) play χ in a full positive game A has the form

$$\begin{array}{cccccccccccccc} \mathsf{P} & & \mathsf{O} & & \mathsf{P} & & \mathsf{O} & & & & \mathsf{P} & & \mathsf{O} & \\ u_0 & \cdot & x_0 & \cdot & u_1 & \cdot & x_1 & \cdot & \cdots & \cdot & u_n & \cdot & x_n & \cdot \quad \cdots \\ \pitchfork & & \pitchfork & & \pitchfork & & \pitchfork & & & & \pitchfork & & \pitchfork & \\ U & & X & & U & & X & & & & U & & X & \end{array}$$

So P plays first (hence the term "*positive*") and all P-moves (resp. O-moves) are available to P (resp. O) when it has to play (hence the term "*full*"). Assuming A is equipped with the winning condition $\mathcal{W}_A$, a play χ as above is *winning* if $(u_k \cdot x_k)_k \in \mathcal{W}_A$.

Consider a uniform automaton $\mathcal{A} : \Sigma$ as in (27), and a Σ-labeled tree T. The *acceptance game* $\mathcal{A}(T)$ is the full positive game with P-moves U and O-moves $X \times \mathfrak{D}$. So a play in $\mathcal{A}(T)$ has the form

$$\begin{array}{cccccccccccccc} \mathsf{P} & & \mathsf{O} & & \mathsf{P} & & \mathsf{O} & & & & \mathsf{P} & & \mathsf{O} & \\ u_0 & \cdot & (x_0, d_0) & \cdot & u_1 & \cdot & (x_1, d_1) & \cdot & \cdots & \cdot & u_n & \cdot & (x_n, d_n) & \cdot \quad \cdots \\ \pitchfork & & \pitchfork & & \pitchfork & & \pitchfork & & & & \pitchfork & & \pitchfork & \\ U & & X \times \mathfrak{D} & & U & & X \times \mathfrak{D} & & & & U & & X \times \mathfrak{D} & \end{array}$$

Similarly as in acceptance games for a usual non-deterministic or alternating automaton (§1.2), O chooses tree directions. Note that if $\mathcal{A}$ is non-deterministic in the sense of Def. 3.1 (*i.e.* $X \simeq \mathbf{1}$), then O only chooses tree directions. Dually, if $\mathcal{A}$ is universal ($U \simeq \mathbf{1}$) then P has no choice. Finally if $\mathcal{A}$ is deterministic ($U \simeq X \simeq \mathbf{1}$) then the only choices available in the game $\mathcal{A}(T)$ are the O's choices of tree directions. Note also that because the sets of P and O-moves of a uniform automaton are always assumed to be non-empty (in this sense uniform automata are always total), there is no maximal finite play in the game $\mathcal{A}(T)$.

We now equip $\mathcal{A}(T)$ with a winning condition $\mathcal{W}_{\mathcal{A}(T)} \subseteq (U \cdot (X \times \mathfrak{D}))^{\omega}$. Each infinite play $\chi = (u_k \cdot (x_k, d_k))_k \in (U \cdot (X \times \mathfrak{D}))^{\omega}$ generates an infinite sequence of states $(q_k)_k \in Q^{\omega}_{\mathcal{A}}$ as follows. We let $q_0 := q^{\iota}_{\mathcal{A}}$ and

$$\begin{array}{lrcl} & q_{k+1} & := & \partial_{\mathcal{A}}(q_k\,,\, \mathsf{a}_k\,,\, u_k\,,\, x_k\,,\, d_k) \\ \text{where} & \mathsf{a}_k & := & T(d_0 \cdot \ldots \cdot d_{k-1}) \end{array}$$

Then χ is winning (*i.e.* $\chi \in \mathcal{W}_{\mathcal{A}(M)}$) iff $(q_k)_k$ is accepting, *i.e.* iff $(q_k)_k \in \Omega_{\mathcal{A}}$. (See also Fig. 2, where states and tree positions are explicitly represented.)

Strategies for P in full positive games are what one expects.

Definition 3.4 (Strategies in Full Positive Games)**.** *A* (P-)strategy *in a full positive game* $A = (U, X)$ *is a function*

$$\sigma \quad : \quad X^* \quad \longrightarrow \quad U$$

Assume now that A *is a game with winning condition* $\mathcal{W}_A$*. Given a strategy* $\sigma : X^* \to U$ *and a sequence* $(x_k)_k \in X^\omega$*, define the sequence* $(u_k)_k \in U^\omega$ *as*

$$u_n \quad := \quad \sigma(x_0 \cdot \ldots \cdot x_{n-1})$$

We then say that σ *is* winning in A *if* $(u_k \cdot x_k)_k \in \mathcal{W}_A$ *for all* $(x_k)_k \in X^\omega$*.*

Example 3.5. *Continuing Ex. 3.2.(ii), given an alternating automaton* $\mathcal{A}$ *over* Σ *and a* Σ*-labeled tree* T*,* P *has a winning strategy in* $\mathcal{A}(T)$ *iff it has a winning strategy in* $\widehat{\mathcal{A}}(T)$*.*

Definition 3.6. *Given a uniform automaton* $\mathcal{A} : \Sigma$ *and a* Σ*-labeled tree* T*, we say that* $\mathcal{A}$ accepts T *if* P *has a winning strategy in* $\mathcal{A}(T)$*, and we let* $\mathcal{L}(\mathcal{A})$ *be the set of* Σ*-labeled trees which are accepted by* $\mathcal{A}$*. Moreover, a set* $\mathcal{L}$ *of* Σ*-labeled trees is* regular *if there is an automaton* $\mathcal{A} : \Sigma$ *such that* $\mathcal{L} = \mathcal{L}(\mathcal{A})$*.*

3.3. Substituted Acceptance Games. We now turn to substituted acceptance games, a simple but central notion of this paper, which allows us to obtain the indexed structure discussed in §2.2. Substituted acceptance games are simply the (essentially obvious) adaptation of the Σ-runs of §2.2 to the acceptance games of §3.2. A similar notion for usual alternating automata was introduced in [Rib15].

Consider a uniform automaton $\mathcal{A} : \Gamma$ as in (27), and a morphism $M \in \mathbf{T}[\Sigma, \Gamma]$. The *uniform substituted acceptance game* $\Sigma \vdash \mathcal{A}(M)$ is the full positive game with P-moves $\Sigma \times U$ and O-moves $X \times \mathfrak{D}$. So a play in $\Sigma \vdash \mathcal{A}(M)$ has the form

$$\begin{array}{ccccccccccccc}
\mathsf{P} & & \mathsf{O} & & \mathsf{P} & & \mathsf{O} & & & & \mathsf{P} & & \mathsf{O} & & \\
(\mathtt{a}_0, u_0) & \cdot & (x_0, d_0) & \cdot & (\mathtt{a}_1, u_1) & \cdot & (x_1, d_1) & \cdot & \cdots & \cdot & (\mathtt{a}_n, u_n) & \cdot & (x_n, d_n) & \cdot & \cdots \\
\Cap & & \Cap & & \Cap & & \Cap & & & & \Cap & & \Cap & & \\
\Sigma \times U & & X \times \mathfrak{D} & & \Sigma \times U & & X \times \mathfrak{D} & & & & \Sigma \times U & & X \times \mathfrak{D} & &
\end{array}$$

Similarly as in a substituted acceptance game for a usual non-deterministic or alternating automaton [Rib15], P chooses input letters and O chooses tree directions. Similarly as in the acceptance games of §3.2, there is no maximal finite play in the game $\Sigma \vdash \mathcal{A}(M)$.

We now equip $\Sigma \vdash \mathcal{A}(M)$ with a winning condition $\mathcal{W}_{\mathcal{A}(M)} \subseteq ((\Sigma \times U) \cdot (X \times \mathfrak{D}))^\omega$. Each infinite play $\chi = ((\mathtt{a}_k, u_k) \cdot (x_k, d_k))_k \in ((\Sigma \times U) \cdot (X \times \mathfrak{D}))^\omega$ generates an infinite sequence of states $(q_k)_k \in Q_{\mathcal{A}}^\omega$ as follows. We let $q_0 := q_{\mathcal{A}}^\imath$ and

$$\begin{array}{rrcl}
 & q_{k+1} & := & \partial_{\mathcal{A}}(q_k\,,\, \mathtt{b}_k\,,\, u_k\,,\, x_k\,,\, d_k) \\
\text{where} & \mathtt{b}_k & := & M(\mathtt{a}_0 \cdot \ldots \cdot \mathtt{a}_k\,,\, d_0 \cdot \ldots \cdot d_{k-1})
\end{array}$$

Then χ is winning (*i.e.* $\chi \in \mathcal{W}_{\mathcal{A}(M)}$) iff $(q_k)_k$ is accepting (*i.e.* iff $(q_k)_k \in \Omega_{\mathcal{A}}$).

Let us set some notations. When the input alphabet Σ is irrelevant or clear from the context, we omit it and write $\mathcal{A}(M)$ for $\Sigma \vdash \mathcal{A}(M)$. We write $\Gamma \vdash \mathcal{A}$ (or simply $\mathcal{A}$) for the game $\Gamma \vdash \mathcal{A}(\mathrm{Id}_\Gamma)$. Moreover, we extend the notation $\mathcal{A}[\mathtt{f}]$ of §2.2 to uniform automata.

	A	$\multimap_{\mathbf{DZ}}$	B	
	$\vdots$		$\vdots$	
O	u			
			v	P
			y	O
P	x			
	$\vdots$		$\vdots$	

Figure 3: A typical zig-zag play with full positive games $A = (U, X)$ and $B = (V, Y)$

Definition 3.7. *Given an ordinary function* $\mathtt{f} : \Sigma \to \Gamma$ *and a uniform automaton* $\mathcal{A} : \Gamma$*, we let* $\mathcal{A}[\mathtt{f}] : \Sigma$ *be the uniform automaton defined as* $\mathcal{A}$*, but with*

$$\partial_{\mathcal{A}[\mathtt{f}]}(q,\, \mathtt{a},\, u,\, x,\, d) \quad := \quad \partial_{\mathcal{A}}(q,\, \mathtt{f}(\mathtt{a}),\, u,\, x,\, d)$$

Similarly as in §2.2, the game $\Sigma \vdash \mathcal{A}([\mathtt{f}])$ is the same as the game $\Sigma \vdash \mathcal{A}[\mathtt{f}]$.

Example 3.8. *Continuing Ex. 3.5, given a usual alternating automaton* $\mathcal{A}$ *over* Γ *and some* $M \in \mathbf{T}[\Sigma, \Gamma]$*,* P *has a winning strategy in* $\mathcal{A}(M)$ *(in the sense of [Rib15]) if and only if* P *has a winning strategy in* $\widehat{\mathcal{A}}(M)$*.*

As expected, substituted acceptance games generalize usual acceptance games. Consider a uniform automaton $\mathcal{A} : \Sigma$ and a Σ-labeled tree T. Let $\dot{T} \in \mathbf{T}[\mathbf{1}, \Sigma]$ be the $\mathbf{T}$-morphism corresponding to T (see §2.2). The game $\mathbf{1} \vdash \mathcal{A}(\dot{T})$ is similar (actually isomorphic in the sense of §3.4) to the acceptance game $\mathcal{A}(T)$ defined in §3.2. A typical play of $\mathbf{1} \vdash \mathcal{A}(\dot{T})$ has the form

$$\begin{array}{ccccccccccccc}
\mathsf{P} & & \mathsf{O} & & \mathsf{P} & & \mathsf{O} & & & & \mathsf{P} & & \mathsf{O} \\
(\bullet, u_0) & \cdot & (x_0, d_0) & \cdot & (\bullet, u_1) & \cdot & (x_1, d_1) & \cdot & \cdots & \cdot & (\bullet, u_n) & \cdot & (x_n, d_n) & \cdot & \cdots \\
\pitchfork & & \pitchfork & & \pitchfork & & \pitchfork & & & & \pitchfork & & \pitchfork & & \\
\mathbf{1} \times U & & X \times \mathfrak{D} & & \mathbf{1} \times U & & X \times \mathfrak{D} & & & & \mathbf{1} \times U & & X \times \mathfrak{D} & &
\end{array}$$

In words, since $\mathbf{1} \simeq \{\bullet\}$ is a singleton, P has actually exactly the same choices in the game $\mathbf{1} \vdash \mathcal{A}(\dot{T})$ as in the game $\mathcal{A}(T)$.

3.4. Zig-Zag Strategies. Zig-zag strategies are at the core of the notion of morphism of our categories of automata. They stem from usual strategies in *simple games* (see e.g. [Abr97, Hyl97]), by imposing a very strong restriction on the shape of plays, essentially corresponding to a form of simulation games. This lead to a very simple notion of strategy, which admits a very simple functional representation (at least compared to usual simple games [BE93, AC98]).

Consider full positive games $A = (U, X)$ and $B = (V, Y)$. Intuitively, a total zig-zag strategy $\sigma : A \multimap_{\mathbf{DZ}} B$ amounts to a strategy for P in an infinite game which consists in countably many sequences of rounds. In a single round $n \in \mathbb{N}$, four moves occur in succession (see also Fig. 3):

(1) O plays a move $u_n \in U$,

(2) P plays a move $v_n \in V$,

(3) O answers with a move $y_n \in Y$,

(4) P concludes with a move $x_n \in X$.

So, in a zig-zag strategy $\sigma : A \multimap_{\mathbf{DZ}} B$, each P-move v_n depends on the previous O-moves $u_0, \ldots, u_n$ and $y_0, \ldots, y_{n-1}$, while each P-move x_n depends on the previous O-moves $u_0, \ldots, u_n$ and $y_0, \ldots, y_{n-1}, y_n$. This leads to the following definition.

Definition 3.9. *Given full positive games $A = (U, X)$ and $B = (V, Y)$, a (total zig-zag)* strategy *$\sigma : A \multimap_{\mathbf{DZ}} B$ is a pair of functions $\sigma = (f, F)$ where*

$$\begin{array}{rcccl} f & : & \bigcup_{n\in\mathbb{N}} \left(U^{n+1} \times Y^n\right) & \longrightarrow & V \\ F & : & \bigcup_{n\in\mathbb{N}} \left(U^{n+1} \times Y^{n+1}\right) & \longrightarrow & X \end{array}$$

Assume now that A and B are equipped with winning conditions $\mathcal{W}_A$ and $\mathcal{W}_B$. Given sequences $(u_n)_n \in U^\omega$ and $(y_n)_n \in Y^\omega$, a strategy σ induces sequences $(v_n)_n \in V^\omega$ and $(x_n)_n \in X^\omega$ defined as

$$\begin{array}{rrcl} & v_n & := & f(u_0 \cdots u_n \,,\, y_0 \cdots y_{n-1}) \in V \\ \textit{and} & x_n & := & F(u_0 \cdots u_n \,,\, y_0 \cdots y_{n-1} \cdot y_n) \in X \end{array}$$

Then σ is winning *if for all $(u_n)_n \in U^\omega$ and all $(y_n)_n \in Y^\omega$, we have $(v_n \cdot y_n)_n \in \mathcal{W}_B$ whenever $(u_n \cdot x_n)_n \in \mathcal{W}_A$.*

It is easy to see that (winning) total zig-zag strategies form a category. We defer the proof of this fact to App. A, which also gives further background on game semantics.

Proposition 3.10. *Full positive games (with winning) and (winning) total zig-zag strategies form a category $\mathbf{DZ}^{(\mathrm{W})}$.*

Remark 3.11. *The functional representation of strategies of [BE93, AC98] is (at least in spirit) inspired from approaches to Gödel's* Dialectica *interpretation (see e.g. [AF98, Koh08]) in categorical logic [dP91] (see also e.g. [HS03, Hyl02, Hof11] and [Jac01, Ex. 1.10.11] for modern refinements and variations). Actually, the category* **DZ** *(for* Dialectica*-like* Zig-zag *games) can be constructed (via a distributive law) from a category of* simple self dualization *[HS99, HS03] (over the topos of trees, see e.g. [BMSS12]), which can be seen as a skeleton of Dialectica-like categories, and our categories of automata (§4) have a shape similar to Dialectica fibrations. Besides, as we shall see in Ex. 6.4, there is an $\exists\forall$-structure on automata which is reminiscent from Gödel's* Dialectica.

The connection between the models presented in this paper and a linear variant of Gödels's Dialectica has been made precise in [PR19] in the case of ω-words.

3.5. Toward Uniform Linear Synchronous Arrow Games. We now prepare to introduce (categories of) uniform linear synchronous arrow games, the last simple but central notion of this paper. Similarly as with §3.3, the material of this section is essentially the adaptation to uniform automata of corresponding notions of [Rib15]. We shall however postpone the proper categorical treatment to §4, as we rely on more advanced material.

Consider substituted acceptance games $\Sigma \vdash \mathcal{A}(M)$ and $\Sigma \vdash \mathcal{B}(N)$. Our goal is to devise a notion of morphism

$$\Sigma \;\vdash\; \mathcal{A}(M) \;\multimap\; \mathcal{B}(N)$$

with a behaviour similar to the *linear synchronous arrow games* of [Rib15]. This would amount to devise a notion of strategy

$$\Sigma \;\vdash\; \sigma \;:\; \mathcal{A}(M) \;\multimap\; \mathcal{B}(N)$$

Σ	$\mathcal{A}(M)$	$\multimap$	$\mathcal{B}(N)$	
	$(\epsilon, \epsilon, q^{\iota}_{\mathcal{A}})$		$(\epsilon, \epsilon, q^{\iota}_{\mathcal{B}})$	
	$\vdots$		$\vdots$	
	$(p, \overline{\mathsf{a}}, q_{\mathcal{A}})$		$(p, \overline{\mathsf{a}}, q_{\mathcal{B}})$	
O	(a, u)			
			(a, v)	P
			(y, d)	O
P	(x, d)			
	$(p.d, \overline{\mathsf{a}}.\mathsf{a}, q'_{\mathcal{A}})$		$(p.d, \overline{\mathsf{a}}.\mathsf{a}, q'_{\mathcal{B}})$	
	$\vdots$		$\vdots$	

Σ	A	$\multimap$	B	
	$\vdots$		$\vdots$	
O	(a, u)			
			v	P
			(y, d)	O
P	x			
	$\vdots$		$\vdots$	

Figure 4: Linear synchronous arrow games

playing similarly as in Fig. 4 (left). Such a strategy σ is therefore required to be a total zig-zag strategy $\mathcal{A}(M) \multimap_{\mathbf{DZ}} \mathcal{B}(N)$ in the sense of Def. 3.9. But in addition, we should as in Fig. 4 also require the $\mathsf{a} \in \Sigma$ and $d \in \mathfrak{D}$ played by P to be the same as their immediate predecessors played by O. The approach we adopt in this paper is to simply remove these moves from the game. This leads to the following notion.

Definition 3.12 (DialZ(Σ)-Games)**.** *Fix an alphabet Σ.*

- *A* DialZ(Σ)*-object A is given by non-empty sets U and X.*
- *Given* DialZ(Σ)*-objects $A = (U, X)$, $B = (V, Y)$, a* DialZ(Σ)*-morphism $\sigma : A \multimap_{\mathsf{DialZ}(\Sigma)} B$ is a total zig-zag strategy (see Fig. 4 (right))*

$$\sigma \quad : \quad (\Sigma \times U, X) \quad \multimap_{\mathbf{DZ}} \quad (V, Y \times \mathfrak{D})$$

Similarly as for **DZ**, the name DialZ stands for *Dialectica*-like *Zig-zag* games. But note the font change. The *sans-serif* categories DialZ(Σ) actually form an indexed category, postponed to §4. This will in particular allow us to equip these games with winning conditions, leading to the indexed category DialAut (for *Dialectica*-like *Automata*).

If we forget about winning, it is possible to see here why DialZ(Σ)-games induce a generalization of the acceptance games of §3.2. First, a substituted acceptance game $\Sigma \vdash \mathcal{A}(M)$ with $\mathcal{A}$ as in (27) induces the DialZ(Σ)-game $A = (U, X)$. Hence substituted acceptance games $\Sigma \vdash \mathcal{A}(M), \mathcal{B}(N)$ induce a DialZ(Σ)-game

$$\mathcal{A}(M) \quad \multimap_{\mathsf{DialZ}(\Sigma)} \quad \mathcal{B}(N)$$

in the obvious way. Consider now an automaton $\mathcal{A} : \Sigma$ as in (27) and a Σ-labeled tree T. As before, let $\dot{T} \in \mathbf{T}[\mathbf{1}, \Sigma]$ be the $\mathbf{T}$-map corresponding to T. Recall the unit automaton $\mathbf{I} : \mathbf{1}$ of Ex. 3.2.(i). Then the moves allowed in $\mathbf{1} \vdash \mathcal{A}(\dot{T})$ correspond exactly to those of the DialZ($\mathbf{1}$)-game $\mathbf{I} \multimap_{\mathsf{DialZ}(\mathbf{1})} \mathcal{A}(\dot{T})$ (see Fig. 5).

4. Fibrations of Tree Automata

In this Section we present an indexed structure for uniform synchronous linear arrow games, in which morphisms $L \in \mathbf{T}[\Delta, \Sigma]$ induce *substitution functors*, and such that the operation $(-)^{\star}$

$\mathbf{1}$	$\mathbf{I}$	$\multimap_{\mathsf{DialZ}(\mathbf{1})}$	$\mathcal{A}(\dot{T})$	
	$(\epsilon, \bullet)$		$(\epsilon, q^{\iota}_{\mathcal{A}})$	
	$\vdots$		$\vdots$	
	$(p, \bullet)$		$(p, q_{\mathcal{A}})$	
O	$(\bullet, \bullet)$			
			u	P
			(x, d)	O
P	$\bullet$			
	$(p.d, \bullet)$		$(p.d, q'_{\mathcal{A}})$	
	$\vdots$		$\vdots$	

Figure 5: Acceptance games as $\mathsf{DialZ}(\mathbf{1})$-games

is itself functorial (see §2.2 and [Jac01, Chap. 1]). While substitution in [Rib15] was defined directly at the level of synchronous arrow games (via the representation of strategies as relations), we devise here an indexed structure induced by a reformulation of synchronicity (in the sense of [Rib15]) based on §3.5. We use the techniques of monoid and comonoid indexing [HS99, HS03] on zig-zag games, which allow for a smooth treatment of monoidal closure and universal quantifications. This will lead us to a proper treatment of the $\mathsf{DialZ}(\Sigma)$-games of Def. 3.12, and to the category $\mathsf{DialAut}$, fibred over $\mathbf{T}$.

The material of this section relies on the symmetric monoidal structure of $\mathbf{DZ}$.

4.1. Symmetric Monoidal Structure of DZ

The category $\mathbf{DZ}$ has a particularly simple symmetric monoidal structure, but which differs from the usual ones in game semantics.

Proposition 4.1. *The category* $\mathbf{DZ}$ *is symmetric monoidal with unit* $\mathbf{I} := (\mathbf{1}, \mathbf{1})$ *and with* $A \otimes B := (U \times V, X \times Y)$ *for* $A = (U, X)$ *and* $B = (V, Y)$.

The action of the tensor $\otimes$ *on strategies* $\sigma_i : A_i \multimap_{\mathbf{DZ}} B_i$ *(for* $i = 1, 2$, $A_i = (U_i, X_i)$ *and* $B_i = (V_i, Y_i)$*) is depicted on Fig. 6. If the* $\sigma_i = (f_i, F_i)$ *where*

$$\begin{array}{rcccl} f_i & : & \bigcup_{n\in\mathbb{N}} \left(U_i^{n+1} \times Y_i^{n}\right) & \longrightarrow & V_i \\ F_i & : & \bigcup_{n\in\mathbb{N}} \left(U_i^{n+1} \times Y_i^{n+1}\right) & \longrightarrow & X_i \end{array}$$

then $\sigma_1 \otimes \sigma_2 = (h, H)$ *where*

$$\begin{array}{rcccl} h & : & \bigcup_{n\in\mathbb{N}} \left((U_1 \times U_2)^{n+1} \times (Y_1 \times Y_2)^{n}\right) & \longrightarrow & V_1 \times V_2 \\ H & : & \bigcup_{n\in\mathbb{N}} \left((U_1 \times U_2)^{n+1} \times (Y_1 \times Y_2)^{n+1}\right) & \longrightarrow & X_1 \times X_2 \end{array}$$

are defined as

$$\begin{array}{rcl} h((\overline{u}_1, \overline{u}_2)\,,\, (\overline{y}_1, \overline{y}_2)) & := & (f_1(\overline{u}_1, \overline{y}_1)\,,\, f_2(\overline{u}_2, \overline{y}_2)) \\ H((\overline{u}_1, \overline{u}_2)\,,\, (\overline{y}_1, \overline{y}_2)) & := & (F_1(\overline{u}_1, \overline{y}_1)\,,\, F_2(\overline{u}_2, \overline{y}_2)) \end{array}$$

The natural structure isomorphisms of $\mathbf{DZ}$ are depicted on Fig. 7. This structure obviously lifts to $\mathbf{DZ}^{\mathrm{W}}$, but we shall not directly use this fact.

	$A_i \xrightarrow{\sigma_i}_{\mathbf{DZ}} B_i$		
	$\vdots$	$\vdots$	
O	u_i		
		v_i	P
		y_i	O
P	x_i		
	$\vdots$	$\vdots$	

	$A_1 \otimes A_2 \xrightarrow{\sigma_1 \otimes \sigma_2}_{\mathbf{DZ}} B_1 \otimes B_2$		
	$\vdots$	$\vdots$	
O	(u_1, u_2)		
		(v_1, v_2)	P
		(y_1, y_2)	O
P	(x_1, x_2)		
	$\vdots$	$\vdots$	

Figure 6: Action of $\otimes$ on $\sigma_i : A_i \multimap_{\mathbf{DZ}} B_i$.

	$(A \otimes B) \otimes C \xrightarrow{\alpha_{A,B,C}}_{\mathbf{DZ}} A \otimes (B \otimes C)$		
	$\vdots$	$\vdots$	
O	$((u, v), w)$		
		$(u, (v, w))$	P
		$(x, (y, z))$	O
P	$((x, y), z)$		
	$\vdots$	$\vdots$	

	$\mathbf{I} \otimes A \xrightarrow{\lambda_A} A$		
	$\vdots$	$\vdots$	
O	$(\bullet, u)$		
		u	P
		x	O
P	$(\bullet, x)$		
	$\vdots$	$\vdots$	

	$A \otimes B \xrightarrow{\gamma_{A,B}}_{\mathbf{DZ}} B \otimes A$		
	$\vdots$	$\vdots$	
O	(u, v)		
		(v, u)	P
		(y, x)	O
P	(x, y)		
	$\vdots$	$\vdots$	

	$A \otimes \mathbf{I} \xrightarrow{\rho_A} A$		
	$\vdots$	$\vdots$	
O	$(u, \bullet)$		
		u	P
		x	O
P	$(x, \bullet)$		
	$\vdots$	$\vdots$	

Figure 7: The structure maps of $\mathbf{DZ}$, for $A = (U, X)$, $B = (V, Y)$ and $C = (W, Z)$

$$
\begin{tikzcd}
(M \otimes M) \otimes M \arrow[r, "\alpha"] \arrow[d, "m \otimes \mathrm{id}_M"'] & M \otimes (M \otimes M) \arrow[r, "\mathrm{id}_M \otimes m"] & M \otimes M \arrow[d, "m"] \\
M \otimes M \arrow[rr, "m"'] & & M
\end{tikzcd}
$$

$$
\begin{tikzcd}
\mathbf{I} \otimes M \arrow[r, "u \otimes \mathrm{id}_M"] \arrow[dr, "\lambda"'] & M \otimes M \arrow[d, "m"] & M \otimes \mathbf{I} \arrow[l, "\mathrm{id}_M \otimes u"'] \arrow[dl, "\rho"] \\
& M &
\end{tikzcd}
\qquad
\begin{tikzcd}
M \otimes M \arrow[rr, "\gamma"] \arrow[dr, "m"'] & & M \otimes M \arrow[dl, "m"] \\
& M &
\end{tikzcd}
$$

Figure 8: Coherence for a monoid (M, m, u) (where α, λ, ρ and γ are symmetric monoidal structure maps)

4.2. Monoid and Comonoid Indexing in DZ

Fix an alphabet Σ. We are now going to see that the $\mathsf{DialZ}(\Sigma)$-games of Def. 3.12 form a category. Recall that given $\mathsf{DialZ}(\Sigma)$-objects $A = (U, X)$ and $B = (V, Y)$, a $\mathsf{DialZ}(\Sigma)$-morphism from A to B is a total zig-zag strategy

$$\sigma \quad : \quad (\Sigma \times U, X) \quad \multimap_{\mathbf{DZ}} \quad (V, Y \times \mathfrak{D})$$

We will use some algebraic structure. Objects of the form $(\mathbf{1}, M)$ (resp. $(K, \mathbf{1})$) are actually (commutative) monoids (resp. comonoids) in $\mathbf{DZ}$. Recall from e.g. [Mel09] that a commutative monoid in a symmetric monoidal category $(\mathbb{C}, \otimes, \mathbf{I})$ is an object M equipped with structure maps $m : M \otimes M \to M$ and $u : \mathbf{I} \to M$ subject to coherence conditions depicted on Fig. 8. A (commutative) comonoid in $\mathbb{C}$ is a (commutative) monoid in $\mathbb{C}^{\mathrm{op}}$. In this paper, by (co)monoid we always mean *commutative* (co)monoid. Write $\mathbf{Comon}(\mathbb{C})$ for the category of comonoids in $\mathbb{C}$. Maps from (K, d, e) to (K', d', e') are $\mathbb{C}$-maps $f : K \to K'$ which commute with the comonoid structure:

$$(f \otimes f) \circ d \quad = \quad d' \circ f \qquad \text{and} \qquad e \quad = \quad e' \circ f \tag{29}$$

It is well-known that the symmetric monoidal structure of $\mathbb{C}$ induces a Cartesian product on $\mathbf{Comon}(\mathbb{C})$ (see e.g. [Mel09, Cor. 18, §6.5]), and conversely that if $(\mathbb{C}, \otimes, \mathbf{I})$ is Cartesian, then every $\mathbb{C}$-object has a canonical comonoid structure. Moreover, note that any set $I \simeq \mathbf{1}$ is a monoid in $\mathbf{Set}$.

Proposition 4.2. *If M, K are non-empty sets and $I \simeq \mathbf{1}$, then $M := (I, M)$ is a monoid and $K := (K, I)$ is a comonoid in $\mathbf{DZ}$. Structure maps are depicted on Fig. 9 (in the case of $I = \mathbf{1}$).*

From now on, we reason modulo the following $\mathbf{DZ}$-isos (with the notations of Prop. 4.2):

$$(\Sigma \times U, X) \quad \simeq \quad \Sigma \otimes (U, X) \qquad \text{and} \qquad (V, Y \times \mathfrak{D}) \quad \simeq \quad (V, Y) \otimes \mathfrak{D}$$

It is well-known (see e.g. [HS99, HS03]) that a monoid M (resp. a comonoid K) in a symmetric monoidal category $(\mathbb{C}, \otimes, \mathbf{I})$ induces a monad $(-) \otimes M$ of indexing with M (resp. a comonad $K \otimes (-)$ of indexing with K).

Proposition 4.3. *Let $(\mathbb{C}, \otimes, \mathbf{I})$ be a symmetric monoidal category.*

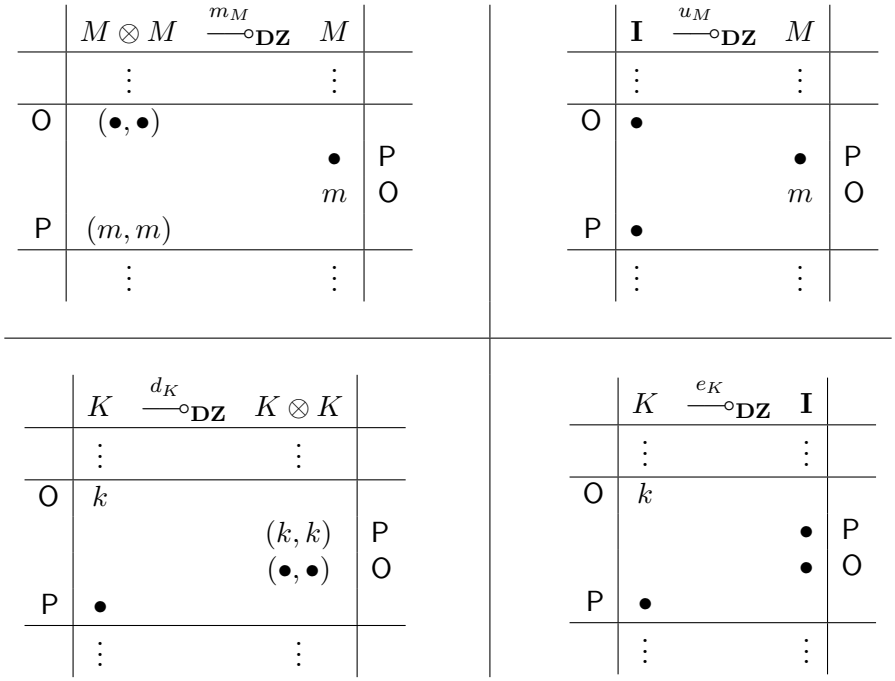


Figure 9: Structure maps for the monoid $M = (\mathbf{1}, M)$ and the comonoid $K = (K, \mathbf{1})$

(a) A monoid (M, m, u) in $\mathbb{C}$ induces a (lax symmetric monoidal) monad $((-) \otimes M, \mu, \eta)$. The functor $(-) \otimes M$ takes an object A to $A \otimes M$ and a map $f : A \to B$ to $f \otimes \mathrm{id}_M : A \otimes M \to B \otimes M$. The natural maps μ and η are given by

$$\begin{array}{lclcrcl} \mu_A & := & (\mathrm{id}_A \otimes m) \circ \alpha & : & (A \otimes M) \otimes M & \longrightarrow & A \otimes M \\ \eta_A & := & (\mathrm{id}_A \otimes u) \circ \rho^{-1} & : & A & \longrightarrow & A \otimes M \end{array}$$

(b) Dually, a comonoid $K = (K, d, e)$ in $\mathbb{C}$ induces an (oplax symmetric monoidal) comonad $(K \otimes (-), \delta, \varepsilon)$, where

$$\begin{array}{lclcrcl} \delta_A & := & \alpha \circ (d \otimes \mathrm{id}_A) & : & K \otimes A & \longrightarrow & K \otimes (K \otimes A) \\ \varepsilon_A & := & \lambda \circ (e \otimes \mathrm{id}_A) & : & K \otimes A & \longrightarrow & A \end{array}$$

The maps ρ, α and λ above are structural isomorphisms of $(\mathbb{C}, \otimes, \mathbf{I})$.

Moreover, the comonad $K \otimes (-)$ is related to the monad $(-) \otimes M$ via a *distributive law*. A distributive law Λ of a comonad (G, δ, ε) over a monad (T, μ, η) on $\mathbb{C}$ is a natural map $\Lambda : G \circ T \Rightarrow T \circ G$ subject to some coherence conditions (see e.g. [HHM07]), which ensure that we have a category $\mathbf{Kl}(\Lambda)$ with the same objects as $\mathbb{C}$ and with homsets

$$\mathbf{Kl}(\Lambda)[A, B] \quad := \quad \mathbb{C}[GA, TB]$$

and that there is a lifting functor $(-)^{\uparrow} : \mathbf{Kl}(\Lambda) \to \mathbb{C}$ taking $f : GA \to TB$ to

$$f^{\uparrow} \quad := \quad G(\mu_B \circ Tf \circ \Lambda_A) \circ \delta_{TA} \quad : \quad GTA \quad \longrightarrow \quad GTB$$

In the case of comonoid and monoid indexing, a distributive law of $K \otimes (-)$ over $(-) \otimes M$ is given by the natural associativity maps:

$$\Phi_{(-)} \quad := \quad \alpha^{-1}_{K,(-),M} \quad : \quad K \otimes ((-) \otimes M) \quad \Longrightarrow \quad (K \otimes (-)) \otimes M$$

Returning to our case, we let

$$\mathsf{DialZ}(\Sigma) \quad := \quad \mathbf{Kl}(\Phi)$$

where Φ is the distributive law of the comonad of indexing with the comonoid Σ over the monad of indexing with the monoid $\mathfrak{D}$ in the category $\mathbf{DZ}$. The canonical lifting functor

$$(-)^{\uparrow} \quad : \quad \mathsf{DialZ}(\Sigma) \quad \longrightarrow \quad \mathbf{DZ}$$

takes a total zig-zag strategy

$$\sigma \quad : \quad \Sigma \otimes A \quad \multimap_{\mathbf{DZ}} \quad B \otimes \mathfrak{D}$$

to a total zig-zag strategy

$$\sigma^{\uparrow} \quad : \quad \Sigma \otimes (A \otimes \mathfrak{D}) \quad \multimap_{\mathbf{DZ}} \quad \Sigma \otimes (B \otimes \mathfrak{D})$$

Modulo associativity, the strategy $\sigma^{\uparrow}$ is given by

$$(\mathrm{id}_{\Sigma} \otimes ((\mathrm{id}_B \otimes m_{\mathfrak{D}}) \circ (\sigma \otimes \mathrm{id}_{\mathfrak{D}}))) \circ (d_{\Sigma} \otimes \mathrm{id}_{A \otimes \mathfrak{D}}) \quad : \quad \Sigma \otimes A \otimes \mathfrak{D} \quad \multimap_{\mathbf{DZ}} \quad \Sigma \otimes B \otimes \mathfrak{D}$$

Note that if σ plays as in Fig. 4 (right) then the strategy

$$\dot{\sigma} \quad := \quad (\mathrm{id}_B \otimes m_{\mathfrak{D}}) \circ (\sigma \otimes \mathrm{id}_{\mathfrak{D}}) \quad : \quad \Sigma \otimes A \otimes \mathfrak{D} \quad \multimap_{\mathbf{DZ}} \quad B \otimes \mathfrak{D}$$

plays as in Fig. 10 (top). It follows that $\sigma^{\uparrow} = (\mathrm{id}_{\Sigma} \otimes \dot{\sigma}) \circ (d_{\Sigma} \otimes \mathrm{id}_{A \otimes \mathfrak{D}})$ plays as in Fig. 10 (bottom).

4.3. The Indexed Structure of $\mathsf{DialZ}(-)$ and the Base Category $\mathbf{T}$

We therefore have for each alphabet Σ a category $\mathsf{DialZ}(\Sigma)$. We now discuss an indexed structure on the categories $\mathsf{DialZ}(-)$, based on a pattern similar to the *simple fibration* $\mathsf{s} : \mathsf{s}(\mathbb{B}) \to \mathbb{B}$ over a category $\mathbb{B}$ with finite products (see e.g. [Jac01, Chap. 1] but also [Hyl02, Hof11]), and reminiscent from [MM15]. The objects of $\mathsf{s}(\mathbb{B})$ are pairs (I, X) of $\mathbb{B}$-objects. The morphisms $(I, X) \to (J, Y)$ are pairs (f_0, f) with $f_0 : I \to J$ and $f : I \times X \to Y$. The functor $\mathsf{s} : \mathsf{s}(\mathbb{B}) \to \mathbb{B}$ is the first projection, and the fibre over I is the Kleisli category of indexing with the comonoid I (see e.g. [Jac01, Ex. 1.3.4]).

A similar construction can be done if instead of a category $\mathbb{B}$ with finite products, one starts from a symmetric monoidal category $\mathbb{C}$, and take as base the category $\mathbf{Comon}(\mathbb{C})$. The fibre over the comonoid K is the Kleisli category $\mathbf{Kl}(K)$ of indexing with K, and a comonoid morphism $u : K \to L$ induces a functor $u^{\star} : \mathbf{Kl}(L) \to \mathbf{Kl}(K)$ acting as the identity on objects and taking $f : L \otimes A \to B$ to $f \circ (u \otimes \mathrm{id}_A) : K \otimes A \to B$. It readily follows that $\mathrm{id}_K^{\star} = \mathrm{id}_{\mathbf{Kl}(K)}$ and that $(u \circ v)^{\star} = v^{\star} \circ u^{\star}$. In other words, we have a functor $\mathbf{Comon}(\mathbb{C})^{\mathrm{op}} \to \mathbf{Cat}$ that we denote $\mathsf{Cl}(\mathbb{C})$ (for *comonoid indexing* over $\mathbb{C}$). The corresponding Grothendieck construction $\int \mathsf{Cl}(\mathbb{C})$ (see e.g. [Jac01, Chap. 1]) is the category whose objects are pairs (K, A) of an object K of $\mathbf{Comon}(\mathbb{C})$ and an object A of $\mathbb{C}$, and whose morphisms from (K, A) to (L, B) are pairs (u, f) where $u : K \to L$ is a comonoid morphism and $f : K \otimes A \to B$. The category $\int \mathsf{Cl}(\mathbb{C})$ is fibred over $\mathbf{Comon}(\mathbb{C})$ via the first projection, that we denote

$$\mathsf{s}_{\mathsf{Cl}}(\mathbb{C}) \quad : \quad \int \mathsf{Cl}(\mathbb{C}) \quad \longrightarrow \quad \mathbf{Comon}(\mathbb{C})$$

Returning to our case, recall that $\mathsf{DialZ}(\Sigma) = \mathbf{Kl}(\Phi)$ where Φ is the distributive law of $\Sigma \otimes (-)$ over $(-) \otimes \mathfrak{D}$. The category $\mathsf{DialZ}(\Sigma)$ can alternatively be obtained as a Kleisli category of

	$\Sigma \otimes A \otimes \mathfrak{D}$	$\xrightarrow{\sigma\otimes\mathrm{id}_{\mathfrak{D}}}_{\mathbf{DZ}}$	$B \otimes \mathfrak{D} \otimes \mathfrak{D}$	$\xrightarrow{\mathrm{id}_B\otimes m_{\mathfrak{D}}}_{\mathbf{DZ}}$	$B \otimes \mathfrak{D}$	
	$\vdots$		$\vdots$		$\vdots$	
O	(a, u)					
			v			
					v	P
					(y, d)	O
			(y, d, d)			
P	(x, d)					
	$\vdots$		$\vdots$		$\vdots$	

	$\Sigma \otimes A \otimes \mathfrak{D}$	$\xrightarrow{d_\Sigma\otimes\mathrm{id}_{A\otimes\mathfrak{D}}}_{\mathbf{DZ}}$	$\Sigma \otimes \Sigma \otimes A \otimes \mathfrak{D}$	$\xrightarrow{\mathrm{id}_\Sigma\otimes\dot{\sigma}}_{\mathbf{DZ}}$	$\Sigma \otimes B \otimes \mathfrak{D}$	
	$\vdots$		$\vdots$		$\vdots$	
O	(a, u)					
			$(\mathsf{a}, \mathsf{a}, u)$			
					(a, v)	P
					(y, d)	O
			(x, d)			
P	(x, d)					
	$\vdots$		$\vdots$		$\vdots$	

Figure 10: Decomposition of $\sigma^{\uparrow}$

	$A \xrightarrow{\sigma} \mathbf{DZ}$	$B \otimes \mathfrak{D}$	
	$\vdots$	$\vdots$	
O	u		
		v	P
		(y,d)	O
P	x		
	$\vdots$	$\vdots$	

	$B \xrightarrow{\tau} \mathbf{DZ}$	$C \otimes \mathfrak{D}$	
	$\vdots$	$\vdots$	
O	v		
		w	P
		(z,d)	O
P	y		
	$\vdots$	$\vdots$	

	$A \xrightarrow{\sigma} \mathbf{DZ}$	$B \otimes \mathfrak{D}$	$\xrightarrow{\tau \otimes \mathrm{id}_{\mathfrak{D}}} \mathbf{DZ}$	$C \otimes \mathfrak{D} \otimes \mathfrak{D}$	$\xrightarrow{\mu_C^{\mathfrak{D}}} \mathbf{DZ}$	$C \otimes \mathfrak{D}$	
	$\vdots$	$\vdots$		$\vdots$		$\vdots$	
O	u						
		v					
				w			
						w	P
						(z,d)	O
				(z,d,d)			
		(y,d)					
P	x						
	$\vdots$	$\vdots$		$\vdots$		$\vdots$	

Figure 11: Composition in $\mathbf{DZ}_{\mathfrak{D}} = \mathbf{Kl}(\mathfrak{D})$

indexing with comonoids over a symmetric monoidal category. Let $\mathbf{DZ}_{\mathfrak{D}}$ be the Kleisli category of indexing with the $\mathbf{DZ}$-monoid $\mathfrak{D}$. The objects of $\mathbf{DZ}_{\mathfrak{D}}$ are full positive games, and maps from A to B are $\mathbf{DZ}$-maps from A to $B \otimes \mathfrak{D}$.

Let us spell out composition in $\mathbf{DZ}_{\mathfrak{D}}$. First recall that for a monad (T, μ, η) on a category $\mathbb{C}$, composition in the Kleisli category $\mathbf{Kl}(T)$ is given by

$$g \circ_{\mathbf{Kl}(T)} f \quad := \quad A \xrightarrow{f} TB \xrightarrow{Tg} TTC \xrightarrow{\mu_C} TC$$

for $f : A \to TB$ and $g : B \to TC$. In the case of $\mathbf{DZ}_{\mathfrak{D}}$-morphisms $\sigma : A \multimap_{\mathbf{DZ}} B \otimes \mathfrak{D}$ and $\tau : B \multimap_{\mathbf{DZ}} C \otimes \mathfrak{D}$ (where $A = (U, X)$, $B = (V, Y)$ and $C = (W, Z)$) as depicted on Fig. 11 (top), the composite $\tau \circ_{\mathbf{DZ}_{\mathfrak{D}}} \sigma$ is depicted (modulo associativity) on Fig. 11 (bottom).

Since $\mathbf{DZ}_{\mathfrak{D}}$ is the Kleisli category of a lax symmetric monoidal monad on $\mathbf{DZ}$, it is symmetric monoidal with structure induced by that of $\mathbf{DZ}$ (see e.g. [Mel09]).

Proposition 4.4. *(a) Consider a monoid M in a symmetric monoidal category $(\mathbb{C}, \otimes, \mathbf{I})$. The Kleisli category $\mathbf{Kl}(M)$ is symmetric monoidal with $A \otimes_{\mathbf{Kl}(M)} B := A \otimes B$ on objects and unit $\mathbf{I}$.*

Moreover, each comonoid (K, d, e) in $\mathbb{C}$ induces a comonoid $(K, \eta^M_{K \otimes K} \circ d, \eta^M_{\mathbf{I}} \circ e)$ in $\mathbf{Kl}(M)$.

(b) In the case of $\mathbf{DZ}_{\mathfrak{D}} = \mathbf{Kl}(\mathfrak{D})$, the action of $\otimes_{\mathbf{DZ}_{\mathfrak{D}}}$ on maps $\sigma_i : A_i \multimap_{\mathbf{DZ}} B_i \otimes \mathfrak{D}$ (for $i = 1, 2$, $A_i = (U_i, X_i)$ and $B_i = (V_i, Y_i)$) is depicted on Fig. 12. If the $\sigma_i = (f_i, F_i)$ where

$$\begin{array}{lcll} f_i & : & \bigcup_{n \in \mathbb{N}} \left(U_i^{n+1} \times Y_i^n \times \mathfrak{D}^n \right) & \longrightarrow \quad V_i \\ F_i & : & \bigcup_{n \in \mathbb{N}} \left(U_i^{n+1} \times Y_i^{n+1} \times \mathfrak{D}^{n+1} \right) & \longrightarrow \quad X_i \end{array}$$

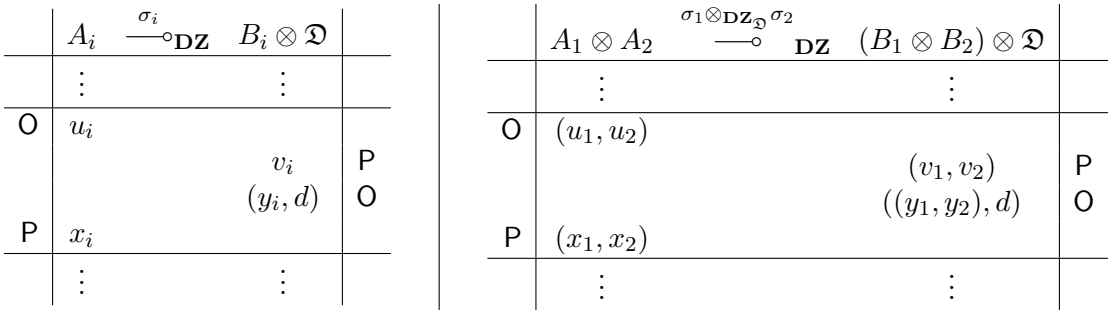


Figure 12: Action of $\otimes_{\mathbf{DZ}_{\mathfrak{D}}}$ on $\sigma_i : A_i \multimap_{\mathbf{DZ}_{\mathfrak{D}}} B_i$

then $\sigma_1 \otimes_{\mathbf{DZ}_{\mathfrak{D}}} \sigma_2 = (h, H)$ *where*

$$\begin{array}{rcll} h & : & \bigcup_{n\in\mathbb{N}} \big((U_1 \times U_2)^{n+1} \times (Y_1 \times Y_2)^n \times \mathfrak{D}^n\big) & \longrightarrow \quad V_1 \times V_2 \\ H & : & \bigcup_{n\in\mathbb{N}} \big((U_1 \times U_2)^{n+1} \times (Y_1 \times Y_2)^{n+1} \times \mathfrak{D}^{n+1}\big) & \longrightarrow \quad X_1 \times X_2 \end{array}$$

are defined as

$$\begin{array}{rcl} h((\overline{u}_1, \overline{u}_2)\,,\, (\overline{y}_1, \overline{y}_2)\,,\, p) & := & (f_1(\overline{u}_1, \overline{y}_1, p)\,,\, f_2(\overline{u}_2, \overline{y}_2, p)) \\ H((\overline{u}_1, \overline{u}_2)\,,\, (\overline{y}_1, \overline{y}_2)\,,\, p) & := & (F_1(\overline{u}_1, \overline{y}_1, p)\,,\, F_2(\overline{u}_2, \overline{y}_2, p)) \end{array}$$

Moreover, the $\mathbf{DZ}_{\mathfrak{D}}$*-structure maps* $\tilde{d}_\Sigma$ *and* $\tilde{e}_\Sigma$ *of the comonoid induced by* Σ *can be depicted as in Fig. 14.*

It follows from Prop. 4.4 and the fact that Φ is a distributive law, that each category $\mathsf{DialZ}(\Sigma)$ is the Kleisli category of indexing with Σ in $\mathbf{DZ}_{\mathfrak{D}}$. We can therefore index $\mathsf{DialZ}(-)$ with the comonoids of $\mathbf{DZ}_{\mathfrak{D}}$. We will actually index $\mathsf{DialZ}(-)$ over the base category $\mathbf{T}$ (of Def. 2.1), which is isomorphic to a full subcategory of $\mathbf{Comon}(\mathbf{DZ}_{\mathfrak{D}})$. First, it directly follows Def. 3.9 that $\mathbf{T}$-strategies from Σ to Γ in the sense of Def. 2.1 correspond exactly to total zig-zag strategies from $(\Sigma, \mathbf{1})$ to $(\Gamma, \mathfrak{D})$, that is to $\mathbf{DZ}_{\mathfrak{D}}$-maps from Σ to Γ. Hence $\mathbf{T}$ is isomorphic to a full subcategory of $\mathbf{DZ}_{\mathfrak{D}}$. Second, $\mathbf{T}$-maps induce comonoid maps.

Proposition 4.5. *The category* $\mathbf{T}$ *embeds in* $\mathbf{Comon}(\mathbf{DZ}_{\mathfrak{D}})$ *via the functor* $\mathsf{E}_{\mathbf{T}}$ *which takes an alphabet* Σ *to the comonoid* $(\Sigma, \tilde{d}_\Sigma, \tilde{e}_\Sigma)$ *and a morphism* $M : \mathbf{T}[\Gamma, \Sigma]$ *to the* $\mathbf{DZ}_{\mathfrak{D}}$*-morphism*

$$\tilde{M} \quad := \quad \jmath_\Sigma \circ M \quad : \quad (\Gamma, \mathbf{1}) \quad \multimap_{\mathbf{DZ}} \quad (\Sigma, \mathbf{1}) \otimes (\mathbf{1}, \mathfrak{D})$$

induced by the $\mathbf{DZ}$*-iso* $\jmath_\Sigma : (\Sigma, \mathfrak{D}) \overset{\simeq}{\multimap}_{\mathbf{DZ}} (\Sigma, \mathbf{1}) \otimes (\mathbf{1}, \mathfrak{D})$.

Proof. We have to check that $\mathbf{T}$-morphisms induce $\mathbf{Comon}(\mathbf{DZ}_{\mathfrak{D}})$-morphisms, that is that the equations (29) hold in $\mathbf{DZ}_{\mathfrak{D}}$:

$$(\tilde{M} \otimes \tilde{M}) \circ \tilde{d}_\Gamma \quad = \quad \tilde{d}_\Sigma \circ \tilde{M} \qquad \text{and} \qquad \tilde{e}_\Gamma \quad = \quad \tilde{e}_\Sigma \circ \tilde{M}$$

Assume that $\tilde{M}$ plays as in Fig. 13 (top left). The first equation follows from the fact that $\tilde{d}_\Sigma \circ \tilde{M}$ plays as in Fig. 13 (middle), while $(\tilde{M} \otimes \tilde{M}) \circ \tilde{d}_\Gamma$ plays as in Fig. 13 (bottom). The second equation follows from the fact that $\tilde{e}_\Sigma \circ \tilde{M}$ plays as in Fig. 13 (top right). $\square$

	Γ	$\xrightarrow{\tilde{M}}_{\mathbf{DZ}_{\mathfrak{D}}}$	Σ	
	$\vdots$		$\vdots$	
O	$\mathtt{b}$			
			$\mathtt{a}$	P
			d	O
P	$\bullet$			
	$\vdots$		$\vdots$	

	Γ	$\xrightarrow{\tilde{M}}_{\mathbf{DZ}_{\mathfrak{D}}}$	Σ	Σ	$\xrightarrow{\tilde{e}_{\Sigma}}_{\mathbf{DZ}_{\mathfrak{D}}}$	$\mathbf{I}$	
	$\vdots$		$\vdots$	$\vdots$		$\vdots$	
O	$\mathtt{b}$						
			$\mathtt{a}$	$\mathtt{a}$			
						$\bullet$	P
						d	O
			d	$\bullet$			
P	$\bullet$						
	$\vdots$		$\vdots$	$\vdots$		$\vdots$	

	Γ	$\xrightarrow{\tilde{M}}_{\mathbf{DZ}_{\mathfrak{D}}}$	Σ	Σ	$\xrightarrow{\tilde{d}_{\Sigma}}_{\mathbf{DZ}_{\mathfrak{D}}}$	$\Sigma \otimes \Sigma$	
	$\vdots$		$\vdots$	$\vdots$		$\vdots$	
O	$\mathtt{b}$						
			$\mathtt{a}$	$\mathtt{a}$			
						$(\mathtt{a}, \mathtt{a})$	P
						d	O
			d	$\bullet$			
P	$\bullet$						
	$\vdots$		$\vdots$	$\vdots$		$\vdots$	

	Γ	$\xrightarrow{\tilde{d}_{\Gamma}}_{\mathbf{DZ}_{\mathfrak{D}}}$	$\Gamma \otimes \Gamma$	$\Gamma \otimes \Gamma$	$\xrightarrow{\tilde{M} \otimes \tilde{M}}_{\mathbf{DZ}_{\mathfrak{D}}}$	$\Sigma \otimes \Sigma$	
	$\vdots$		$\vdots$	$\vdots$		$\vdots$	
O	$\mathtt{b}$						
			$(\mathtt{b}, \mathtt{b})$	$(\mathtt{b}, \mathtt{b})$			
						$(\mathtt{a}, \mathtt{a})$	P
						d	O
			d	$\bullet$			
P	$\bullet$						
	$\vdots$		$\vdots$	$\vdots$		$\vdots$	

Figure 13: **T**-maps as comonoids morphisms in the proof of Prop. 4.5

	Σ	$\xrightarrow{\tilde{d}_\Sigma}{}_{\mathbf{DZ}_{\mathfrak{D}}}$	$\Sigma \otimes \Sigma$	
	$\vdots$		$\vdots$	
O	a			
			(a, a)	P
			d	O
P	$\bullet$			
	$\vdots$		$\vdots$	

	Σ	$\xrightarrow{\tilde{e}_\Sigma}{}_{\mathbf{DZ}_{\mathfrak{D}}}$	$\mathbf{I}$	
	$\vdots$		$\vdots$	
O	a			
			$\bullet$	P
			d	O
P	$\bullet$			
	$\vdots$		$\vdots$	

Figure 14: Structure maps for the comonoid $\Sigma = (\Sigma, \mathbf{1})$

We thus get an indexed category

$$\mathsf{DialZ} \quad := \quad \mathsf{Cl}(\mathbf{DZ}_{\mathfrak{D}}) \circ \mathsf{E}_{\mathbf{T}} \quad : \quad \mathbf{T}^{\mathrm{op}} \quad \longrightarrow \quad \mathbf{Cat}$$

We already mentioned the well-known fact that the symmetric monoidal structure of a category induces a Cartesian structure on its category of comonoids (see e.g. [Mel09, Cor. 18, §6.5]). By Prop. 4.5, this gives a Cartesian structure on $\mathbf{T}$.

Corollary 4.6. *The category* $\mathbf{T}$ *is Cartesian, with on objects the Cartesian product of alphabets, and with unit* $\mathbf{1}$.

4.4. The Fibred Category DialAut

We thus have a category DialZ indexed over $\mathbf{T}$, and whose fibre over Σ is the category $\mathsf{DialZ}(\Sigma)$. We will now define a fibration $\mathsf{da} : \mathsf{DialAut} \to \mathbf{T}$ (for *Dialectica*-like *Automata*) of uniform substituted acceptance games, which essentially extends DialZ with winning (and acceptance). The fibration $\mathsf{da} : \mathsf{DialAut} \to \mathbf{T}$ is obtained by applying the Grothendieck construction to an indexed category $(-)^\star : \mathbf{T}^{\mathrm{op}} \to \mathbf{Cat}$, which takes an alphabet Σ to a category $\mathsf{DialAut}_\Sigma$. The action of $(-)^\star$ on $\mathbf{T}$-maps is based on the indexed category DialZ.

Definition 4.7 (The Category $\mathsf{DialAut}_\Sigma$)**.** *Fix an alphabet* Σ.

- *The objects of the category* $\mathsf{DialAut}_\Sigma$ *are tuples* $(U, X, \mathcal{W}_A)$ *where* U *and* X *are non-empty sets and where* $\mathcal{W}_A \subseteq ((\Sigma \times U) \cdot (X \times \mathfrak{D}))^\omega$.
- *The* $\mathsf{DialAut}_\Sigma$*-morphisms from* $(U, X, \mathcal{W}_A)$ *to* $(V, Y, \mathcal{W}_B)$ *are* $\mathsf{DialZ}(\Sigma)$*-morphisms from* (U, X) *to* (V, Y)*, that is total zig-zag strategies*

$$\sigma \quad : \quad \Sigma \otimes (U, X) \quad \longrightarrow_{\mathbf{DZ}} \quad (V, Y) \otimes \mathfrak{D}$$

whose lift $\sigma^\uparrow$ *are winning strategies on*

$$(\Sigma \times U, X \times \mathfrak{D}, \mathcal{W}_A) \quad \longrightarrow_{\mathbf{DZ}^{\mathrm{w}}} \quad (\Sigma \times V, Y \times \mathfrak{D}, \mathcal{W}_B)$$

Composition and identities of $\mathsf{DialAut}_\Sigma$ are induced by composition and identities of $\mathsf{DialZ}(\Sigma)$ (using the functoriality of $(-)^\uparrow$ for winning). Given a uniform automaton $\mathcal{A} : \Delta$ and $M \in \mathbf{T}[\Sigma, \Delta]$, we still write $\Sigma \vdash \mathcal{A}(M)$ for the $\mathsf{DialAut}_\Sigma$-object induced by the uniform substituted acceptance game $\Sigma \vdash \mathcal{A}(M)$ of §3.5.

We now turn to substitution and indexing. Morphisms $L \in \mathbf{T}[\Gamma, \Sigma]$ induce functors

$$L^\star \quad : \quad \mathsf{DialAut}_\Sigma \quad \longrightarrow \quad \mathsf{DialAut}_\Gamma$$

defined as follows. Given a $\mathsf{DialAut}_\Sigma$-object $A = (U, X, \mathcal{W}_A)$, we let $L^\star(A)$ be the $\mathsf{DialAut}_\Gamma$-object $(U, X, L^\star(\mathcal{W}_A))$, where

$$((\mathtt{b}_k, u_k) \cdot (x_k, d_k))_k \in L^\star(\mathcal{W}_A) \quad \text{iff} \quad ((L(\mathtt{b}_0. \cdots .\mathtt{b}_k,\, d_0. \cdots .d_{k-1}), u_k) \cdot (x_k, d_k))_k \in \mathcal{W}_A$$

When the $\mathsf{DialAut}_\Sigma$-object A is induced by a uniform substituted acceptance game $\Sigma \vdash \mathcal{A}(M)$, we have the expected result that $L^\star(A)$ is induced by the uniform substituted acceptance game $\Gamma \vdash \mathcal{A}(M \circ L)$ (see §2.2).

Lemma 4.8. *Given a uniform substituted acceptance game $\Sigma \vdash \mathcal{A}(M)$ and $L \in \mathbf{T}[\Gamma, \Sigma]$, we have*

$$L^\star(\Sigma \vdash \mathcal{A}(M)) \quad = \quad \Gamma \vdash \mathcal{A}(M \circ L)$$

Proof. Recall from §3.2 that $\mathcal{W}_{\mathcal{A}(M)} \subseteq ((\Sigma \times U) \cdot (X \times \mathfrak{D}))^\omega$ is the set of infinite plays $((\mathtt{a}_k, u_k) \cdot (x_k, d_k))_k \in ((\Sigma \times U) \cdot (X \times \mathfrak{D}))^\omega$ such that $(q_k)_k \in \Omega_\mathcal{A}$, where $q_0 = q^\imath_\mathcal{A}$ and

$$q_{k+1} \quad = \quad \partial_\mathcal{A}(q_k\,,\, M(\mathtt{a}_0. \cdots .\mathtt{a}_k\,,\, d_0. \cdots .d_{k-1})\,,\, u_k\,,\, x_k\,,\, d_k)$$

Now, we have $((\mathtt{b}_k, u_k) \cdot (x_k, d_k))_k \in L^\star(\mathcal{W}_{\mathcal{A}(M)})$ if and only if

$$((L(\mathtt{b}_0. \cdots .\mathtt{b}_k,\, d_0. \cdots .d_{k-1}), u_k) \cdot (x_k, d_k))_k \in \mathcal{W}_{\mathcal{A}(M)}$$

that is, if and only if $(q_k)_k \in \Omega_\mathcal{A}$ for the sequence of states $(q_k)_k$ with $q_0 = q^\imath_\mathcal{A}$ and

$$\begin{array}{rrcl} & q_{k+1} & := & \partial_\mathcal{A}(q_k\,,\, M(\mathtt{a}_0. \cdots .\mathtt{a}_k\,,\, d_0. \cdots .d_{k-1})\,,\, u_k\,,\, x_k\,,\, d_k) \\ \text{where} & \mathtt{a}_\ell & := & L(\mathtt{b}_0. \cdots .\mathtt{b}_\ell\,,\, d_0. \cdots .d_{\ell-1}) \end{array}$$

But for $\mathtt{a}_0. \cdots .\mathtt{a}_k$ with

$$\mathtt{a}_\ell \quad = \quad L(\mathtt{b}_0. \cdots .\mathtt{b}_\ell\,,\, d_0. \cdots .d_{\ell-1})$$

we have

$$M(\mathtt{a}_0 \cdot \ldots \cdot \mathtt{a}_k\,,\, d_0 \cdot \ldots \cdot d_{k-1}) \quad = \quad (M \circ L)(\mathtt{b}_0 \cdot \ldots \cdot \mathtt{b}_k\,,\, d_0 \cdot \ldots \cdot d_{k-1})$$

so that the sequence of state $(q_k)_k$ actually satisfies

$$q_{k+1} \quad = \quad \partial_\mathcal{A}(q_k\,,\, (M \circ L)(\mathtt{b}_0 \cdot \ldots \cdot \mathtt{a}_k\,,\, d_0 \cdot \ldots \cdot d_{k-1})\,,\, u_k\,,\, x_k\,,\, d_k)$$

We thus get

$$((\mathtt{b}_k, u_k) \cdot (x_k, d_k))_k \in L^\star(\mathcal{W}_{\mathcal{A}(M)}) \quad \text{iff} \quad ((\mathtt{b}_k, u_k) \cdot (x_k, d_k))_k \in \mathcal{W}_{\mathcal{A}(M \circ L)}$$

□

The action of $L^\star$ on maps is induced by $\mathsf{Cl}(\mathbf{DZ}_\mathfrak{D})(L) : \mathsf{DialZ}(\Sigma) \to \mathsf{DialZ}(\Gamma)$, so that for $\sigma \in \mathsf{DialAut}_\Sigma[A, B]$, we let

$$L^\star(\sigma) \quad := \quad \sigma \circ (L \otimes \mathrm{id}_A)$$

(where $\circ$, $\otimes$ and id_A are taken in $\mathbf{DZ}_\mathfrak{D}$). It remains to check that $L^\star(\sigma)^\uparrow$ is winning whenever so is $\sigma^\uparrow$. Our proof relies on the representation of strategies as sets of plays (App. A).

Proposition 4.9. *Let $L \in \mathbf{T}[\Gamma, \Sigma]$ and consider* $\mathsf{DialAut}_\Sigma$*-objects $A = (U, X, \mathcal{W}_A)$ and $B = (V, Y, \mathcal{W}_B)$. Given a total strategy $\sigma : \Sigma \otimes (U, X) \multimap_{\mathbf{DZ}} (V, Y) \otimes \mathfrak{D}$, if the strategy $\sigma^\uparrow$ is winning on*

$$(\Sigma \times U, X \times \mathfrak{D}, \mathcal{W}_A) \quad \multimap_{\mathbf{DZ}} \quad (\Sigma \times V, Y \times \mathfrak{D}, \mathcal{W}_B)$$

then the strategy $L^\star(\sigma)^\uparrow$ is winning on

$$(\Gamma \times U, X \times \mathfrak{D}, L^\star(\mathcal{W}_A)) \quad \multimap_{\mathbf{DZ}} \quad (\Gamma \times V, Y \times \mathfrak{D}, L^\star(\mathcal{W}_B))$$

Proof. First note that for an arbitrary total zig-zag strategy $\tau : C \multimap_{\mathbf{DZ}} D$ (for full positive games C and D), every infinite play χ such that $\exists^\infty k.\chi(0).\cdots.\chi(k) \in \tau$ is uniquely determined by $\chi_{\restriction C}$ and $\chi_{\restriction D}$. In the following, we write $\chi = (\chi_{\restriction C}, \chi_{\restriction D})$.

Assume that σ plays as in Fig. 4 (right) so that (reasoning as in §4.2) $\sigma^\uparrow$ plays as in Fig. 10 (bottom). Hence, if $L \in \mathbf{T}[\Gamma, \Sigma]$ is represented by the strategy depicted on Fig. 15 (top left), then modulo associativity $L^\star(\sigma)$ plays as in Fig. 15 (bottom) so that $L^\star(\sigma)^\uparrow$ plays as in Fig. 15 (top right).

Consider now an infinite play χ of $L^\star(\sigma)^\uparrow$, that is an infinite play χ on

$$(\Gamma \times U, X \times \mathfrak{D}) \quad \multimap \quad (\Gamma \times V, Y \times \mathfrak{D})$$

such that $\exists^\infty k.\chi(0).\cdots.\chi(k) \in L^\star(\sigma)^\uparrow$. Write

$$\chi \quad = \quad (((\mathsf{b}_k, u_k) \cdot (x_k, d_k))_k \ , \ ((\mathsf{b}_k, v_k) \cdot (y_k, d_k))_k)$$

so that $(((\mathsf{b}_k, u_k) \cdot x_k)_k \, , \, (v_k \cdot (y_k, d_k))_k)$ is an infinite play of $L^\star(\sigma)$ and

$$(((L(\mathsf{b}_0.\cdots.\mathsf{b}_k,\, d_0.\cdots.d_{k-1}), u_k) \cdot x_k)_k \ , \ (v_k \cdot (y_k, d_k))_k)$$

is an infinite play of σ. But it follows that

$$((L(\mathsf{b}_0.\cdots.\mathsf{b}_k,\, d_0.\cdots.d_{k-1}), u_k) \cdot (x_k, d_k))_k \in \mathcal{W}_A \quad \Longrightarrow \quad ((L(\mathsf{b}_0.\cdots.\mathsf{b}_k,\, d_0.\cdots.d_{k-1}), v_k) \cdot (y_k, d_k))_k \in \mathcal{W}_B$$

and by definition of the action of $L^\star$ on $\mathsf{DialAut}_\Sigma$-objects, we thus get

$$((\mathsf{b}_k, u_k) \cdot (x_k, d_k))_k \in L^\star(\mathcal{W}_A) \quad \Longrightarrow \quad ((\mathsf{b}_k, v_k) \cdot (y_k, d_k))_k \in L^\star(\mathcal{W}_B)$$

□

We thus obtain an indexed category $(-)^\star : \mathbf{T}^{\mathrm{op}} \to \mathbf{Cat}$ since $(-)^\star$ is itself functorial. We let $\mathsf{da} : \mathsf{DialAut} \to \mathbf{T}$ be obtained by applying the Grothendieck construction to $(-)^\star$.

Definition 4.10 (The Fibred Category $\mathsf{DialAut}$)**.** *The objects of* $\mathsf{DialAut}$ *are pairs (Σ, A) where A is an object of* $\mathsf{DialAut}_\Sigma$*. Maps from (Σ, A) to (Γ, B) are pairs (L, σ) of a $\mathbf{T}$-map $L : \Sigma \to \Gamma$ and a* $\mathsf{DialAut}_\Sigma$*-map σ from A to $L^\star(B)$.*

The fibration

$$\mathsf{da} \quad : \quad \mathsf{DialAut} \quad \longrightarrow \quad \mathbf{T}$$

is the first projection, so that $\mathsf{da}(\Sigma, A) := \Sigma$ *and* $\mathsf{da}(L, \sigma) := L$.

	$\Gamma \xrightarrow{L} \mathbf{DZ}_{\mathfrak{D}}$	Σ	
	$\vdots$	$\vdots$	
O	$\mathtt{b}$		
		$\mathtt{a}$	P
		d	O
P	$\bullet$		
	$\vdots$	$\vdots$	

	$(\Gamma \times U, X \times \mathfrak{D}) \xrightarrow{L^\star(\sigma)^\uparrow}$	$(\Gamma \times V, Y \times \mathfrak{D})$	
	$\vdots$	$\vdots$	
O	$(\mathtt{b}, u)$		
		$(\mathtt{b}, v)$	P
		(y, d)	O
P	(x, d)		
	$\vdots$	$\vdots$	

	$\Gamma \otimes A \xrightarrow{L \otimes_{\mathbf{DZ}_{\mathfrak{D}}} \mathrm{id}_A^{\mathbf{DZ}_{\mathfrak{D}}}} \mathbf{DZ}$	$\Sigma \otimes A \otimes \mathfrak{D} \xrightarrow{(\mathrm{id}_B \otimes m) \circ (\sigma \otimes \mathrm{id}_{\mathfrak{D}})} \mathbf{DZ}$	$B \otimes \mathfrak{D}$	
	$\vdots$	$\vdots$	$\vdots$	
O	$(\mathtt{b}, u)$			
		$(\mathtt{a}, u)$		
			v	P
			(y, d)	O
		(x, d)		
P	x			
	$\vdots$	$\vdots$	$\vdots$	

Figure 15: The strategies L, $L^\star(\sigma)$ and $L^\star(\sigma)^\uparrow$ of the proof of Prop. 4.9

4.5. Substitution and Language Inclusion

We now check that $\mathsf{DialAut}_\Sigma$ is correct w.r.t. language inclusion. First, consider substituted acceptance games $\Sigma \vdash \mathcal{A}(M)$ and $\Sigma \vdash \mathcal{B}(N)$ in the sense of §3.3. We thus obtain $\mathsf{DialAut}_\Sigma$ objects, that we still write $\Sigma \vdash \mathcal{A}(M), \mathcal{B}(N)$. Now, it follows from Lem. 4.8 that given

$$\sigma \ : \ \mathcal{A}(M) \ \multimap \ \mathcal{B}(N) \qquad \text{and} \qquad L \in \mathbf{T}[\Gamma, \Sigma]$$

we have

$$L^\star(\sigma) \ : \ \mathcal{A}(M \circ L) \ \multimap \ \mathcal{B}(N \circ L)$$

Hence, $\mathsf{DialAut}$ interprets all instances of the (Subst) rule (15) of the form

$$\frac{M \ ; \ \mathcal{A} \vdash \mathcal{B}}{M \circ L \ ; \ \mathcal{A} \vdash \mathcal{B}} \quad (\text{where } M \in \mathbf{T}[\Sigma, \Delta] \text{ and } L \in \mathbf{T}[\Gamma, \Sigma])$$

In particular, given $\mathcal{A}, \mathcal{B} : \Sigma$, for all Σ-labeled tree T (and using the notation of §2.2.(b)) we have

$$\frac{\Sigma \ ; \ \mathcal{A} \vdash \mathcal{B}}{\dot{T} \ ; \ \mathcal{A} \vdash \mathcal{B}}$$

Assume given $\sigma : \mathcal{A} \multimap \mathcal{B}$. If $T \in \mathcal{L}(\mathcal{A})$, then there is some $\tau : \mathbf{I_1} \multimap \mathcal{A}(T)$. It follows that we obtain $\dot{T}^\star(\sigma) \circ \tau : \mathbf{I_1} \multimap \mathcal{B}(T)$, which implies $T \in \mathcal{L}(\mathcal{B})$. In other words, $\sigma : \mathcal{A} \multimap \mathcal{B}$ and T induce a function

$$\tau \ : \ \mathbf{I} \multimap \mathcal{A}(T) \qquad \longmapsto \qquad T^\star(\sigma) \circ \tau \ : \ \mathbf{I} \multimap \mathcal{B}(T)$$

and we have shown:

Proposition 4.11 (Thm. 1.3, (9))**.** *If* P *has a winning strategy in* $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$*, then we have* $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})$*.*

5. Symmetric Monoidal Closed Structure

We show here that the category $\mathbf{DZ}$ of full positive games and total zig-zag strategies is equipped with a monoidal closed structure, and that this structure lifts to $\mathsf{DialZ}(\Sigma)$ and to the fibres of $\mathsf{DialAut}$. This in particular gives a (symmetric) monoidal closed structure on uniform automata.

We first discuss the closed structure of $\mathbf{DZ}$ (§5.1). We then show how the symmetric monoidal closed structure of $\mathbf{DZ}$ lifts to $\mathsf{DialAut}$ and to uniform tree automata (§5.2). This provides a realizability interpretation of a propositional linear (multiplicative) deduction system (§5.3). We finally show how the closed structure gives a (functorial) notion of linear complement (§5.4).

Recall from e.g. [Mel09] that a symmetric monoidal category $(\mathbb{C}, \otimes, \mathbf{I})$ is *closed* if for every object A, the functor $A \otimes (-)$ has a right adjoint $(-)^A$. According to [ML98, Thm. IV.1.2], it is sufficient to show that for every object C there is an object C^A and map

$$\mathsf{eval}_C \ : \ A \otimes C^A \ \longrightarrow \ C$$

such that for every $f : A \otimes B \to C$ there is a unique $\mathbf{\Lambda}(f) : B \to C^A$ with

$$\begin{tikzcd} A \otimes C^A \arrow[r, "\mathsf{eval}_C"] & C \\ A \otimes B \arrow[u, dotted, "\mathrm{id}_A \otimes \mathbf{\Lambda}(f)"] \arrow[ur, "f"'] & \end{tikzcd}$$

5.1. The Symmetric Monoidal Closure of DZ

The monoidal closed structure of **DZ** can actually be read off from the definition of zig-zag strategies given in Def. 3.9.

Let us see how to define a linear exponent full positive game $B^A = (A \multimap_{\mathbf{DZ}} B)$ from full positive games $A = (U, X)$ and $B = (V, Y)$, such that a strategy $\sigma : A \longrightarrow_{\mathbf{DZ}} B$ induces (modulo $A \simeq A \otimes \mathbf{I}$) a strategy $\mathbf{\Lambda}(\sigma) : \mathbf{I} \longrightarrow_{\mathbf{DZ}} (A \multimap_{\mathbf{DZ}} B)$. Assume that σ plays as in Fig. 3. From each play s of σ, the responses $v \in V$ of σ to O-moves $u \in U$ define a function

$$f_s \quad : \quad U \quad \longrightarrow \quad V$$

and the responses $x \in X$ of σ to further O-moves $y \in Y$ define a function

$$F_s \quad : \quad U \times Y \quad \longrightarrow \quad X$$

This amounts to describe σ by a pair of maps

$$\begin{array}{rcccl} f & : & \bigcup_{n \in \mathbb{N}} (U^n \times Y^n) & \longrightarrow & (U \longrightarrow V) \\ F & : & \bigcup_{n \in \mathbb{N}} (U^n \times Y^n) & \longrightarrow & (U \times Y \longrightarrow X) \end{array} \tag{30}$$

Proposition 5.1. *The category* **DZ** *is symmetric monoidal closed. The linear exponent of* $A = (U, X)$ *and* $B = (V, Y)$ *is* $A \multimap_{\mathbf{DZ}} B \ := \ (V^U \times X^{U \times Y}, U \times Y)$.

The monoidal closed structure of **DZ** departs from traditional game semantics since the natural isomorphism $A \otimes B \longrightarrow_{\mathbf{DZ}} C \simeq B \longrightarrow_{\mathbf{DZ}} (A \multimap_{\mathbf{DZ}} C)$ relates strategies, but not *plays*.

Proof of Prop. 5.1. We use notations of App. A. Let $A = (U, X)$, $B = (V, Y)$ and $C = (W, Z)$, so that $A \multimap_{\mathbf{DZ}} C$ is the game $(W^U \times X^{U \times Z}, U \times Z)$. The total zig-zag strategy eval_C is defined as follows:

	$A \otimes (A \multimap_{\mathbf{DZ}} C)$ $\overset{\mathsf{eval}_C}{\longrightarrow}_{\mathbf{DZ}}$	C	
	$\vdots$	$\vdots$	
O	$(u, (f, F))$		
		$f(u)$	P
		z	O
P	$(F(u, z), (u, z))$		
	$\vdots$	$\vdots$	

Consider first the uniqueness requirement of monoidal closure. Given any total

$$\tau' \quad : \quad B \quad \longrightarrow_{\mathbf{DZ}} \quad (A \multimap_{\mathbf{DZ}} C)$$

the composite $\mathsf{eval}_C \circ (\mathrm{id}_A \otimes \tau')$ plays as in Fig. 16. It follows that $\tau' = \tau''$ whenever $\mathsf{eval}_C \circ (\mathrm{id}_A \otimes \tau') = \mathsf{eval}_C \circ (\mathrm{id}_A \otimes \tau'')$, since any distinct pairs (f', F') and (f'', F'') can be distinguished with O-moves $u \in U$ and $z \in Z$.

Fix now some total zig-zag $\sigma : A \otimes B \longrightarrow_{\mathbf{DZ}} C$. We define

$$\tau \quad = \quad \mathbf{\Lambda}(\sigma) \quad : \quad B \quad \longrightarrow_{\mathbf{DZ}} \quad (A \multimap_{\mathbf{DZ}} C)$$

by induction on plays. To each $(s, t) \in \mathrm{HS}(\tau)$, with s and t even-length, we associate $(s', t') \in \mathrm{HS}(\sigma)$, with s' and t' of the same length, and such that, for $(\overline{v}, \overline{y}) = \partial(s)$ and $((\overline{f}, \overline{F}), (\overline{u}, \overline{z})) = \partial(t)$, we have $\partial(s') = ((\overline{u}, \overline{v}), (\overline{F}(\overline{u}, \overline{z}), \overline{y}))$ and $\partial(t') = (\overline{f}(\overline{u}), \overline{z})$, where we take the pointwise application of sequences of functions.

	$A \otimes B$	$\xrightarrow{\mathrm{id}_A \otimes \tau'}_{\mathbf{DZ}}$	$A \otimes (A \multimap_{\mathbf{DZ}} C)$	$\xrightarrow{\mathsf{eval}_C}_{\mathbf{DZ}}$	C	
	$\vdots$		$\vdots$		$\vdots$	
O	(u, v)					
			$(u, (f', F'))$			
					$f'(u)$	P
					z	O
			$(F'(u, z), (u, z))$			
P	$(F'(u, z), y')$					
	$\vdots$		$\vdots$		$\vdots$	

Figure 16: The composite $\mathsf{eval}_C \circ (\mathrm{id}_A \otimes \tau')$ in the proof of Prop. 5.1

- For the base case, we put $(\epsilon, \epsilon) \in \mathrm{HS}(\tau)$, and associate it to $(\epsilon, \epsilon) \in \mathrm{HS}(\sigma)$.
- Assume now $(s, t) \in \mathrm{HS}(\tau)$, associated to $(s', t') \in \mathrm{HS}(\sigma)$. For each $v \in V$, we define the functions $f_v : U \to W$ and $F_v : U \times Z \to X$ as follows: given $u \in U$, let w such that $(s'.(u, v), t'.w) \in \mathrm{HS}(\sigma)$, and for each $z \in Z$, let x and $y_{u,z}$ such that $(s'.(u, v).(x, y_{u,z}), t'.w.z) \in \mathrm{HS}(\sigma)$. We then let $f_v(u) := w$ and $F_v(u, z) := x$. We now let $(s.v.y_{u,z}, t.(f_v, F_v).(u, z)) \in \mathrm{HS}(\tau)$, and associate it to $(s'.(u, v).(x, y_{u,z}), t'.w.z) = (s'.(u, v).(F_v(u, z), y_{u,z}), t'.f_v(u).z)$ so that the invariant is satisfied.

It then follows from the invariant that we indeed have $\mathsf{eval}_C \circ \mathrm{id}_A \otimes \tau = \sigma$. First note that the map $(s, t) \in \mathrm{HS}(\tau) \mapsto (s', t') \in \mathrm{HS}(\sigma)$ is surjective. The property then follows from the fact that $(s, t) \in \mathrm{HS}(\tau)$ iff $(s', t') \in \mathrm{HS}(\mathsf{eval}_C \circ (\mathrm{id}_A \otimes \tau))$. This is shown by induction on pairs of plays $(s, t) \in \wp_B^{\mathsf{even}} \times \wp_{A \multimap_{\mathbf{DZ}} C}^{\mathsf{even}}$. The base case is trivial. For the induction step, given

$$(s \cdot v \cdot y_{u,z}\ ,\ t \cdot (f_v, F_v) \cdot (u, z)) \quad \in \quad \wp_B^{\mathsf{even}} \times \wp_{A \multimap_{\mathbf{DZ}} C}^{\mathsf{even}}$$

we have $(s \cdot v \cdot y_{u,z}\ ,\ t \cdot (f_v, F_v) \cdot (u, z)) \in \mathrm{HS}(\tau)$ if and only if

$$(s' \cdot (u, v) \cdot (F_v(u, z), y_{u,z})\ ,\ t' \cdot f_v(u) \cdot z) \quad \in \quad \mathrm{HS}(\mathsf{eval}_C \circ (\mathrm{id}_A \otimes \tau))$$

and we are done. □

5.2. The Symmetric Monoidal Closed Structure of DialAut and Tree Automata

The symmetric monoidal closed structure of DialAut and of tree automata is induced by the symmetric monoidal closed structure of DialZ, which is itself lifted from **DZ**.

5.2.1. The Symmetric Monoidal Structure of DialZ. We have seen in Prop. 4.4 that the symmetric monoidal structure of **DZ** lifts via monoid indexing to give a symmetric monoidal structure to $\mathbf{DZ}_{\mathfrak{D}}$. The same scheme actually applies to DialZ, which is symmetric monoidal with structure induced by comonoid indexing in $\mathbf{DZ}_{\mathfrak{D}}$.

Proposition 5.2. *(a) Consider a comonoid K in a symmetric monoidal category $\mathbb{C}$. The Kleisli category $\mathbf{Kl}(K)$ is symmetric monoidal with $A \otimes_{\mathbf{Kl}(K)} B := A \otimes B$ on objects and unit $\mathbf{I}$.*

	$\Sigma \otimes A_i$ $\overset{\sigma_i}{\longrightarrow}_{\mathbf{DZ}_{\mathfrak{D}}}$	B_i
	$\vdots$	$\vdots$
O	(a, u_i)	
P		v_i
O		(y_i, d)
P	x_i	
	$\vdots$	$\vdots$

	$\Sigma \otimes (A_1 \otimes A_2)$ $\overset{\tau}{\longrightarrow}_{\mathbf{DZ}_{\mathfrak{D}}}$	$B_1 \otimes B_2$	
	$\vdots$	$\vdots$	
O	$(\mathsf{a}, (u_1, u_2))$		
		(v_1, v_2)	P
		$((y_1, y_2), d)$	O
P	(x_1, x_2)		
	$\vdots$	$\vdots$	

Figure 17: Action of $\otimes_{\mathsf{DialZ}(\Sigma)}$ on $\sigma_i : A_i \multimap_{\mathsf{DialZ}(\Sigma)} B_i$, where $\tau := \sigma_1 \otimes_{\mathsf{DialZ}(\Sigma)} \sigma_2$

(b) In the case of $\mathsf{DialZ}(\Sigma) = \mathbf{Kl}(\Sigma)$*, the action of the tensor* $\otimes_{\mathsf{DialZ}(\Sigma)}$ *on strategies* $\sigma_i : \Sigma \otimes A_i \longrightarrow_{\mathbf{DZ}_{\mathfrak{D}}} B_i$ *(for* $i = 1, 2$*,* $A_i = (U_i, X_i)$ *and* $B_i = (V_i, Y_i)$*) is depicted on Fig. 17. If the* $\sigma_i = (f_i, F_i)$ *where*

$$\begin{array}{rcccl} f_i & : & \bigcup_{n\in\mathbb{N}} \left(\Sigma^{n+1} \times U_i^{n+1} \times Y_i^n \times \mathfrak{D}^n\right) & \longrightarrow & V_i \\ F_i & : & \bigcup_{n\in\mathbb{N}} \left(\Sigma^{n+1} \times U_i^{n+1} \times Y_i^{n+1} \times \mathfrak{D}^{n+1}\right) & \longrightarrow & X_i \end{array}$$

then $\sigma_1 \otimes_{\mathsf{DialZ}(\Sigma)} \sigma_2 = (h, H)$ *where*

$$\begin{array}{rcccl} h & : & \bigcup_{n\in\mathbb{N}} \left(\Sigma^{n+1} \times (U_1 \times U_2)^{n+1} \times (Y_1 \times Y_2)^n \times \mathfrak{D}^n\right) & \longrightarrow & V_1 \times V_2 \\ H & : & \bigcup_{n\in\mathbb{N}} \left(\Sigma^{n+1} \times (U_1 \times U_2)^{n+1} \times (Y_1 \times Y_2)^{n+1} \times \mathfrak{D}^{n+1}\right) & \longrightarrow & X_1 \times X_2 \end{array}$$

are defined as

$$\begin{array}{rcl} h(\overline{\mathsf{a}}\,,\, (\overline{u}_1, \overline{u}_2)\,,\, (\overline{y}_1, \overline{y}_2)\,,\, p) & := & (f_1(\overline{\mathsf{a}}, \overline{u}_1, \overline{y}_1, p)\,,\, f_2(\overline{\mathsf{a}}, \overline{u}_2, \overline{y}_2, p)) \\ H(\overline{\mathsf{a}}\,,\, (\overline{u}_1, \overline{u}_2)\,,\, (\overline{y}_1, \overline{y}_2)\,,\, p) & := & (F_1(\overline{\mathsf{a}}, \overline{u}_1, \overline{y}_1, p)\,,\, F_2(\overline{\mathsf{a}}, \overline{u}_2, \overline{y}_2, p)) \end{array}$$

5.2.2. The Symmetric Monoidal Closure of $\mathbf{DZ}_{\mathfrak{D}}$ and DialZ. The monoidal closed structure of $\mathbf{DZ}$ lifts to $\mathbf{DZ}_{\mathfrak{D}}$ and to the fibres of DialZ. In the case of $\mathbf{DZ}_{\mathfrak{D}}$, since

$$\mathbf{DZ}_{\mathfrak{D}}[A \otimes B\ ,\ C] \quad = \quad \mathbf{DZ}[A \otimes B\ ,\ C \otimes \mathfrak{D}] \quad \simeq \quad \mathbf{DZ}[A\ ,\ (B \multimap_{\mathbf{DZ}} C \otimes \mathfrak{D})]$$

we should have $(A \multimap_{\mathbf{DZ}_{\mathfrak{D}}} B) \otimes \mathfrak{D} \simeq (A \multimap_{\mathbf{DZ}} B \otimes \mathfrak{D})$. Given $A = (U, X)$ and $B = (V, Y)$ this leads to $(A \multimap_{\mathbf{DZ}_{\mathfrak{D}}} B) = (W, Z)$ with

$$(W, Z \times \mathfrak{D}) \quad \simeq \quad (V^U \times X^{U \times Y \times \mathfrak{D}}\ ,\ U \times Y \times \mathfrak{D})$$

We therefore let

$$(U, X) \multimap_{\mathbf{DZ}_{\mathfrak{D}}} (V, Y) \quad := \quad (V^U \times X^{U \times Y \times \mathfrak{D}}\ ,\ U \times Y)$$

The closed structure of $\mathbf{DZ}_{\mathfrak{D}}$ directly lifts to $\mathsf{DialZ}(\Sigma)$ since

$$\mathsf{DialZ}(\Sigma)[A \otimes B\ ,\ C] \quad = \quad \mathbf{DZ}_{\mathfrak{D}}[\Sigma \otimes (A \otimes B)\ ,\ C] \quad \simeq \quad \mathbf{DZ}_{\mathfrak{D}}[\Sigma \otimes A\ ,\ B \multimap_{\mathbf{DZ}_{\mathfrak{D}}} C]$$

Proposition 5.3. *The categories* $\mathbf{DZ}_{\mathfrak{D}}$ *and* $\mathsf{DialZ}(\Sigma)$ *are symmetric monoidal closed.*

5.2.3. The Symmetric Monoidal Closed Structure of DialAut**.** The symmetric monoidal closed structure of DialZ gives the fibrewise symmetric monoidal closed structure of DialAut (in the sense of [Jac01, §1.8]). The unit over Σ is $\mathbf{I}_\Sigma := (\mathbf{1}, \mathbf{1}, \mathbf{1}^\omega)$. Given $\mathsf{DialAut}_\Sigma$-objects $A = (U, X, \mathcal{W}_A)$ and $B = (V, Y, \mathcal{W}_B)$, let

$$\begin{array}{rcl} A \otimes_{\mathsf{DA}} B & := & (U \times V \ , \ X \times Y \ , \ \mathcal{W}_A \sqcap \mathcal{W}_B) \\ A \multimap_{\mathsf{DA}} B & := & (V^U \times X^{U \times Y \times \mathfrak{D}} \ , \ U \times Y \ , \ \mathcal{W}_A \sqcup \mathcal{W}_B) \end{array}$$

with

$$\varpi \in \mathcal{W}_A \sqcap \mathcal{W}_B \quad \text{iff} \quad \big(\varpi_{\restriction (\Sigma \times U) + (X \times \mathfrak{D})} \in \mathcal{W}_A \quad \text{and} \quad \varpi_{\restriction (\Sigma \times V) + (Y \times \mathfrak{D})} \in \mathcal{W}_B\big)$$

and

$$((\mathsf{a}_k \, , \, (f_k, F_k)) \cdot ((u_k, y_k) \, , \, d_k))_k \ \in \ \mathcal{W}_A \sqcup \mathcal{W}_B \quad \text{iff} \quad (\alpha \in \mathcal{W}_A \quad \Longrightarrow \quad \beta \in \mathcal{W}_B)$$

where α and β are obtained by pointwise application:

$$\begin{array}{rcl} \alpha & := & ((\mathsf{a}_k \, , \, u_k) \cdot (F_k(u_k, y_k, d_k) \, , \, d_k))_k \\ \beta & := & ((\mathsf{a}_k \, , \, f_k(u_k)) \cdot (y_k \, , \, d_k))_k \end{array}$$

In the notations $A \otimes_{\mathsf{DA}} B$ and $A \multimap_{\mathsf{DA}} B$ we omit the subscript DA and write $A \otimes B$ and $A \multimap B$ whenever possible.

Proposition 5.4. *The fibration* DialAut *is fibrewise monoidal closed.*

5.2.4. The Symmetric Monoidal Closed Structure of Uniform Automata. We now turn to uniform automata. Their symmetric monoidal closed structure is inherited from $\mathsf{DialAut}_\Sigma$.

Definition 5.5 (Monoidal Product and Linear Implication on Uniform Automata)**.** *Assume $\mathcal{A}$ is as in (27) and*

$$\mathcal{B} \quad = \quad (Q_\mathcal{B} \, , \, q^\imath_\mathcal{B} \, , \, V \, , \, Y \, , \, \partial_\mathcal{B} \, , \, \Omega_\mathcal{B})$$

so that

$$\begin{array}{rrcccccc} & \partial_\mathcal{A} & : & Q_\mathcal{A} \times \Sigma & \longrightarrow & U \times X & \longrightarrow & (\mathfrak{D} \longrightarrow Q_\mathcal{A}) \\ \textit{and} & \partial_\mathcal{B} & : & Q_\mathcal{B} \times \Sigma & \longrightarrow & V \times Y & \longrightarrow & (\mathfrak{D} \longrightarrow Q_\mathcal{B}) \end{array}$$

- *We let $\mathcal{A} \otimes \mathcal{B}$ be the automaton over Σ defined as*

$$\mathcal{A} \otimes \mathcal{B} \quad := \quad (Q_\mathcal{A} \times Q_\mathcal{B} \, , \, (q^\imath_\mathcal{A}, q^\imath_\mathcal{B}) \, , \, U \times V \, , \, X \times Y \, , \, \partial_{\mathcal{A} \otimes \mathcal{B}} \, , \, \Omega_{\mathcal{A} \otimes \mathcal{B}})$$

  with

$$\partial_{\mathcal{A} \otimes \mathcal{B}}((q_\mathcal{A}, q_\mathcal{B}) \, , \, \mathsf{a} \, , \, (u, v) \, , \, (x, y) \, , \, d) \quad := \quad (q'_\mathcal{A} \, , \, q'_\mathcal{B})$$

  where

$$q'_\mathcal{A} \ := \ \partial_\mathcal{A}(q_\mathcal{A} \, , \, \mathsf{a} \, , \, u \, , \, x \, , \, d) \qquad \textit{and} \qquad q'_\mathcal{B} \ := \ \partial_\mathcal{B}(q_\mathcal{B} \, , \, \mathsf{a} \, , \, v \, , \, y \, , \, d)$$

  and with $((q_n, q'_n))_n \in \Omega_{\mathcal{A} \otimes \mathcal{B}}$ iff $((q_n)_n \in \Omega_\mathcal{A}$ and $(q'_n)_n \in \Omega_\mathcal{B})$.

- *We let $(\mathcal{A} \multimap \mathcal{B})$ be the automaton over Σ defined as*

$$(\mathcal{A} \multimap \mathcal{B}) \quad := \quad (Q_\mathcal{A} \times Q_\mathcal{B} \, , \, (q^\imath_\mathcal{A}, q^\imath_\mathcal{B}) \, , \, V^U \times X^{U \times Y \times \mathfrak{D}} \, , \, U \times Y \, , \, \partial_{\mathcal{A} \multimap \mathcal{B}} \, , \, \Omega_{\mathcal{A} \multimap \mathcal{B}})$$

  with

$$\partial_{\mathcal{A} \multimap \mathcal{B}}((q_\mathcal{A}, q_\mathcal{B}) \, , \, \mathsf{a} \, , \, (f, F) \, , \, (u, y) \, , \, d) \quad := \quad (q'_\mathcal{A} \, , \, q'_\mathcal{B})$$

where

$$q'_{\mathcal{A}} \quad = \quad \partial_{\mathcal{A}}(q_{\mathcal{A}}\,,\,\mathtt{a}\,,\,u\,,\,F(u,y,d)\,,\,d) \qquad \textit{and} \qquad q'_{\mathcal{B}} \quad = \quad \partial_{\mathcal{B}}(q_{\mathcal{B}}\,,\,\mathtt{a}\,,\,f(u)\,,\,y\,,\,d)$$

and with $((q_n, q'_n))_n \in \Omega_{\mathcal{A}\multimap\mathcal{B}}$ *iff* $((q_n)_n \in \Omega_{\mathcal{A}} \implies (q'_n)_n \in \Omega_{\mathcal{B}})$.

Note that $\Omega_{\mathcal{A}\otimes\mathcal{B}}$ as well as $\Omega_{\mathcal{A}\multimap\mathcal{B}}$ are ω-regular since $\Omega_{\mathcal{A}}$ and $\Omega_{\mathcal{B}}$ are both assumed to be ω-regular. Note also that $\mathcal{A}\otimes\mathcal{B}$ is non-deterministic (resp. universal, deterministic) if both $\mathcal{A}$ and $\mathcal{B}$ are non-deterministic (resp. universal, deterministic). Moreover, assuming $\mathcal{A},\mathcal{B} : \Gamma$ and $M \in \mathbf{T}[\Sigma,\Gamma]$, we have, as $\mathsf{DialAut}_\Sigma$-objects,

$$\begin{array}{rrcl} & \Sigma \vdash (\mathcal{A}(M) \multimap_{\mathsf{DA}} \mathcal{B}(M)) & \simeq & \Sigma \vdash (\mathcal{A} \multimap \mathcal{B})(M) \\ \text{and} & \Sigma \vdash (\mathcal{A}(M) \otimes_{\mathsf{DA}} \mathcal{B}(M)) & \simeq & \Sigma \vdash (\mathcal{A} \otimes \mathcal{B})(M) \end{array}$$

Note also that we obtain a notion of linear complement with

$$\mathcal{A}^{\perp} \quad := \quad \mathcal{A} \multimap \perp$$

where $\perp$ is a particular automaton accepting no tree (see §5.4), and it follows from monoidal closure that $(-)^{\perp}$ is a contravariant functor taking $\sigma : \mathcal{A}(M) \multimap \mathcal{B}(N)$ to $\sigma^{\perp} : \mathcal{B}^{\perp}(N) \multimap \mathcal{A}^{\perp}(M)$.

5.3. Deduction, Adequacy and Correctness

We now return to the deduction system for automata outlined in §1, and discuss the role of linearity in our setting following §2.4 and §2.5. First, the monoidal structure of $\mathsf{DialAut}_\Sigma$ allows us to interpret sequents of the form

$$M \;;\; \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B} \tag{14}$$

where $M \in \mathbf{T}[\Sigma,\Gamma]$ and $\mathcal{A}_1,\ldots,\mathcal{A}_n,\mathcal{B}$ are uniform automata over Γ. The sequent (14) is interpreted as the homset

$$\mathsf{DialAut}_\Sigma[\mathcal{A}_1(M) \otimes_{\mathsf{DA}} \cdots \otimes_{\mathsf{DA}} \mathcal{A}_n(M)\;,\;\mathcal{B}(M)]$$

Moreover, the monoidal closed structure implies that (14) can equivalently be interpreted as the set of winning P-strategies in the uniform substituted acceptance game

$$\Sigma \vdash (\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B})(M)$$

Second, the symmetric monoidal closed structure allows us to interpret the deduction rules of IMLL. We gather them on Fig. 18. Using the notations of §2.2, we write $\mathcal{A}_1,\ldots,\mathcal{A}_n \vdash \mathcal{B}$ to denote the sequent $\mathrm{Id}\;;\;\mathcal{A}_1,\ldots,\mathcal{A}_n \vdash \mathcal{B}$. Our model is sound w.r.t. this deduction system.

Proposition 5.6 (Adequacy)**.** *If the sequent* $M \;;\; \mathcal{A}_1,\ldots,\mathcal{A}_n \vdash \mathcal{B}$ *is derivable using the rules of Fig. 18, then there is a winning* P*-strategy* σ *in*

$$\mathcal{A}_1(M) \otimes_{\mathsf{DA}} \cdots \otimes_{\mathsf{DA}} \mathcal{A}_n(M) \quad \longrightarrow \quad \mathcal{B}(M)$$

In particular, if $\mathcal{A} \vdash \mathcal{B}$ is derivable, then by combining Prop. 5.6 with Prop. 4.11, we obtain a strategy witnessing that $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})$.

$$(\textsc{Exchange})\ \frac{M\ ;\ \overline{\mathcal{A}},\ \mathcal{A},\ \mathcal{B},\ \overline{\mathcal{C}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\ \mathcal{B},\ \mathcal{A},\ \overline{\mathcal{C}} \vdash \mathcal{C}}$$

$$(\textsc{Cut})\ \frac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\ ;\ \overline{\mathcal{B}},\ \mathcal{A},\ \overline{\mathcal{C}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{B}},\ \overline{\mathcal{A}},\ \overline{\mathcal{C}} \vdash \mathcal{C}} \qquad \frac{}{M\ ;\ \mathcal{A} \vdash \mathcal{A}}\ (\textsc{Axiom})$$

$$(\textsc{Left}\ \otimes)\ \frac{M\ ;\ \overline{\mathcal{A}},\ \mathcal{A},\ \mathcal{B},\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\ \mathcal{A} \otimes \mathcal{B},\ \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\ ;\ \overline{\mathcal{B}} \vdash \mathcal{B}}{M\ ;\ \overline{\mathcal{A}},\ \overline{\mathcal{B}} \vdash \mathcal{A} \otimes \mathcal{B}}\ (\textsc{Right}\ \otimes)$$

$$(\textsc{Left}\ \mathbf{I})\ \frac{M\ ;\ \overline{\mathcal{A}},\ \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}},\ \mathbf{I},\ \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{}{M\ ;\ \vdash \mathbf{I}}\ (\textsc{Right}\ \mathbf{I})$$

$$(\textsc{Left}\ \multimap)\ \frac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A} \qquad M\ ;\ \overline{\mathcal{B}},\ \mathcal{B},\ \overline{\mathcal{C}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{B}},\ \overline{\mathcal{A}},\ \mathcal{A} \multimap \mathcal{B},\ \overline{\mathcal{C}} \vdash \mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{A}},\ \mathcal{B} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{B} \multimap \mathcal{C}}\ (\textsc{Right}\ \multimap)$$

Figure 18: Rules of IMLL for uniform automata

Proof of Prop. 5.6. The strategy σ is built (as usual) by induction on the derivation $\mathscr{D}$ of the sequent $M\ ;\ \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B}$, and using the categorical combinator corresponding to the last applied rule.

- If $\mathscr{D}$ ends with the (Axiom) rule, then σ is the identity (copy-cat) strategy.
- If $\mathscr{D}$ ends with the (Cut) rule, then σ is obtained (using symmetric monoidal structure) by composing the strategies obtained by induction hypothesis for the left and right premises.
- If $\mathscr{D}$ ends either with the (Exchange) rule, or a rule for $\otimes$ or $\mathbf{I}$, then σ is obtained using the fibrewise symmetric monoidal structure of DialAut.
- If $\mathscr{D}$ ends with a rule for $\multimap$, then σ is obtained from the induction hypothesis using the fibrewise monoidal closure of DialAut. □

Note that the strategy σ is obtained from the derivation $\mathscr{D}$ in a purely compositional way. Moreover, all the rules of Fig. 18 are compatible with cut-elimination.

Remark 5.7 (On Cut-Elimination)**.** *It follows from the fact that we have monoidal closed categories (Prop. 5.4), that the interpretation of derivations as strategies for the rules of Fig. 18 is compatible with cut-elimination, in the sense that if a derivation $\mathscr{D}'$ is obtained from a derivation $\mathscr{D}$ by applying the proof transformation steps described in e.g. [Mel09, §3.3], then $\mathscr{D}$ and $\mathscr{D}'$ are interpreted by the same strategy. This in particular applies to the following two derivations:*

$$\dfrac{\dfrac{\dfrac{\mathscr{D}_1}{\mathcal{A} \vdash \mathcal{B}}}{\mathbf{I} \vdash \mathcal{A} \multimap \mathcal{B}} \qquad \dfrac{\dfrac{\mathscr{D}_2}{\mathbf{I} \vdash \mathcal{A}} \qquad \dfrac{}{\mathcal{B} \vdash \mathcal{B}}}{\mathcal{A} \multimap \mathcal{B} \vdash \mathcal{B}}}{\mathbf{I} \vdash \mathcal{B}} \qquad \dfrac{\begin{array}{c}\vdots \\ \mathscr{D}_1[\mathscr{D}_2/\mathcal{A}]\end{array}}{\mathbf{I} \vdash \mathcal{B}}$$

Example 5.8. *Proposition 5.6 yields a winning* P*-strategy in*

$$\mathcal{B} \otimes \mathcal{B} \otimes (\mathcal{B} \multimap \mathcal{A}) \quad \longrightarrow \quad \mathcal{A} \otimes \mathcal{B}$$

obtained from the proof tree

$$\dfrac{\dfrac{\dfrac{\overline{\mathcal{B} \vdash \mathcal{B}} \qquad \overline{\mathcal{A} \vdash \mathcal{A}}}{\mathcal{B}, \mathcal{B} \multimap \mathcal{A} \vdash \mathcal{A}} \qquad \overline{\mathcal{B} \vdash \mathcal{B}}}{\mathcal{B}, (\mathcal{B} \multimap \mathcal{A}), \mathcal{B} \vdash \mathcal{A} \otimes \mathcal{B}}}{\mathcal{B}, \mathcal{B}, (\mathcal{B} \multimap \mathcal{A}) \vdash \mathcal{A} \otimes \mathcal{B}}$$

Note that in Fig. 18 we omitted the *weakening* and *contraction* rules (22):

$$(\textsc{Weak})\ \dfrac{M\ ;\ \overline{\mathcal{A}}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}}, \mathcal{A}, \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad\qquad \dfrac{M\ ;\ \overline{\mathcal{A}}, \mathcal{A}, \mathcal{A}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}}, \mathcal{A}, \overline{\mathcal{B}} \vdash \mathcal{C}}\ (\textsc{Contr})$$

Similarly as with usual automata, the contraction rule can be interpreted on *non-deterministic* uniform automata but not on general uniform automata. This rule amounts to providing winning P-strategies in the game

$$\mathcal{A} \quad \multimap \quad \mathcal{A} \otimes \mathcal{A} \tag{31}$$

If $\mathcal{A}$ is non-deterministic (and with P-moves U), then a winning P-strategy in (31) simply takes an O-move $u \in U$ in component $\mathcal{A}$ to the pair $(u, u) \in U \times U$ in component $\mathcal{A} \otimes \mathcal{A}$. Note that such strategy may not exist when $\mathcal{A}$ is a general uniform automaton, that is when it is equipped with a set of O-moves $X \not\simeq \mathbf{1}$, since O can play two different $(x, x') \in X \times X$ in the component $\mathcal{A} \otimes \mathcal{A}$, that P may not be able to merge into a single $x'' \in X$ in the left component $\mathcal{A}$.

On the other hand, the weakening rule, which asks for a winning P-strategy in

$$\mathcal{A} \quad \multimap \quad \mathbf{I}$$

can always be realized (since we required the set of P and O-moves to be always non-empty), but in a non-canonical way for general uniform automata. More generally, given $\mathcal{A}$ and $\mathcal{B}$ over the same input alphabet, there is always a winning P-strategy in

$$\mathcal{A} \otimes \mathcal{B} \quad \multimap \quad \mathcal{A} \tag{32}$$

Assuming $\mathcal{A}$ and $\mathcal{B}$ are as in Def. 5.5, such a strategy takes $(u, v) \in U \times V$ to $u \in U$ and takes $x \in X$ to $(x, y) \in X \times Y$, where y is an arbitrarily chosen element of Y.

We shall come back on the connection between non-deterministic automata, the interpretation of the (Weak) and (Contr) rules and IMELL in §7.

Example 5.9. *Proposition 5.6 actually holds for any extension of the deduction system of Fig. 18 with realizable rules, that is with rules*

$$\overline{\mathcal{A} \vdash \mathcal{B}}$$

such that there is a winning P*-strategy in* $\mathcal{A} \multimap \mathcal{B}$*. In particular:*

(i) We can extend the system with the following generalization of (32):

$$\overline{\mathcal{A}_1\,, \ldots, \mathcal{A}_n \vdash \mathcal{A}_i}$$

We thus get

$$\dfrac{\dfrac{\dfrac{\overline{\mathcal{A} \vdash \mathcal{A}} \qquad \overline{\mathcal{B} \vdash \mathcal{B}}}{\mathcal{A}, \mathcal{B} \vdash \mathcal{A} \otimes B} \qquad \overline{\mathcal{A} \otimes \mathcal{B} \vdash \mathcal{A}}}{\mathcal{A}, \mathcal{B} \vdash \mathcal{A}}}{\mathcal{A} \vdash \mathcal{B} \multimap \mathcal{A}}$$

So there is a winning P*-strategy on*

$$\mathcal{A} \quad \multimap \quad (\mathcal{B} \multimap \mathcal{A})$$

and by Prop. 4.11 we have

$$\mathcal{L}(\mathcal{A}) \quad \subseteq \quad \mathcal{L}(\mathcal{B} \multimap \mathcal{A})$$

(ii) For $\mathcal{B}$ non-deterministic, we can extend the system with the following generalizations of (31):

$$\frac{}{\mathcal{B} \vdash \mathcal{B} \otimes \ldots \otimes \mathcal{B}}$$

Continuing Ex. 5.8 with $\mathcal{B}$ non-deterministic, we thus have

$$\dfrac{\dfrac{}{\mathcal{B} \vdash \mathcal{B} \otimes \mathcal{B}} \qquad \dfrac{\dfrac{\vdots}{\mathcal{B}, \mathcal{B}, (\mathcal{B} \multimap \mathcal{A}) \vdash \mathcal{A} \otimes \mathcal{B}}}{\mathcal{B} \otimes \mathcal{B}, (\mathcal{B} \multimap \mathcal{A}) \vdash \mathcal{A} \otimes \mathcal{B}}}{\mathcal{B}, (\mathcal{B} \multimap \mathcal{A}) \vdash \mathcal{A} \otimes \mathcal{B}}$$

Finally, we note that the monoidal structure together with (32) implies that $\otimes$ indeed implements a conjunction on automata.

Proposition 5.10. *Given $\mathcal{A}, \mathcal{B} : \Sigma$, we have $\mathcal{L}(\mathcal{A} \otimes \mathcal{B}) = \mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B})$.*

Proof. The inclusion ($\subseteq$) is given by winning strategies in $\mathcal{A} \otimes \mathcal{B} \multimap \mathcal{A}$ and $\mathcal{A} \otimes \mathcal{B} \multimap \mathcal{B}$.

For the other direction, using Prop. 5.4, tensor σ winning on $\mathbf{I_1} \multimap \mathcal{A}(T)$ with τ winning on $\mathbf{I_1} \multimap \mathcal{B}(T)$ and then precompose with a monoidal unit map. □

5.4. Falsity and Complementation

We have already seen in §1.2 that usual alternating automata are equipped with a complementation construction $(-)^{\perp\!\!\!\perp}$ linear in the number of states (see e.g. [MS87]). Using the monoidal closed structure, a similar construction can be done with uniform automata.

Definition 5.11 (Falsity Uniform Automaton)**.** *For each alphabet Σ, the* falsity uniform automaton $\lightning$ *over Σ is*

$$\lightning \quad := \quad (\mathbb{B}\,,\, \mathbb{f}\,,\, \mathfrak{D}\,,\, \mathbf{1}\,,\, \partial_{\lightning}\,,\, \Omega_{\lightning})$$

where $\Omega_{\lightning} := \mathbb{B}^* \cdot \mathbb{t}^\omega$ and where

$$\partial_{\lightning}(\mathbb{b}\,,\, _\,,\, d'\,,\, \bullet\,,\, d) \quad := \quad \begin{cases} \mathbb{f} & \textit{if } \mathbb{b} = \mathbb{f} \textit{ and } d = d' \\ \mathbb{t} & \textit{otherwise} \end{cases}$$

Note that in the game $\Sigma \vdash \lightning$, O looses as soon as it does not play the same tree direction as proposed by P. On the other hand, $\lightning$ accepts no tree since in an acceptance game $\lightning(T)$, O can always play the same d as P.

Consider a uniform automaton $\mathcal{A} : \Sigma$ with set of P-moves U and set of O-moves X. The automaton $(\mathcal{A} \multimap \lightning)$ is isomorphic (via $X^{U \times \mathfrak{D}} \simeq X^{U \times \mathbf{1} \times \mathfrak{D}}$) to the automaton $\mathcal{A}^{\lightning}$ defined as

$$\mathcal{A}^{\lightning} \quad := \quad (Q_{\mathcal{A}} \times \mathbb{B}\,,\, (q^{\imath}_{\mathcal{A}}, \mathbb{f})\,,\, \mathfrak{D}^U \times X^{U \times \mathfrak{D}}\,,\, U\,,\, \partial_{\mathcal{A}^{\lightning}}\,,\, \Omega_{\mathcal{A}^{\lightning}})$$

where

$$(q_k, \mathbb{b}_k)_k \in \Omega_{\mathcal{A}^{\lightning}} \qquad \text{iff} \qquad ((q_k)_k \in \Omega_{\mathcal{A}} \quad \Longrightarrow \quad (\mathbb{b}_k)_k \in \mathbb{B}^* \cdot \mathbb{t}^\omega)$$

and where

$$\partial_{\mathcal{A}^{\perp\!\!\!\perp}}(\mathsf{a}\,,\,(q_{\mathcal{A}},\mathbb{b})\,,\,(f,F)\,,\,u\,,\,d) \quad := \quad \begin{cases} (q'_{\mathcal{A}}\,,\,\mathbb{f}) & \text{if } \mathbb{b}=\mathbb{f} \text{ and } d=f(u) \\ (q'_{\mathcal{A}}\,,\,\mathbb{t}) & \text{otherwise} \end{cases}$$

with $q'_{\mathcal{A}} := \partial_{\mathcal{A}}(\mathsf{a},\, q_{\mathcal{A}},\, u,\, F(u,d),\, d)$. Hence O looses as soon as it does not follow the direction proposed by P via f.

Thanks to the determinacy of ω-regular games (see e.g. [Tho97, PP04]), we get:

Proposition 5.12. *Given $\mathcal{A} : \Sigma$, we have $\mathcal{L}(\mathcal{A}^{\perp\!\!\!\perp}) = \Sigma^{\mathfrak{D}^*} \setminus \mathcal{L}(\mathcal{A})$.*

Proof. The argument is an adaptation of the one given in [Wal02]. By determinacy of ω-regular games, it is equivalent to show that:

$$\mathsf{P} \text{ wins the game } \mathbf{1} \vdash \mathcal{A}^{\perp\!\!\!\perp}(\dot{T}) \qquad \Longleftrightarrow \qquad \mathsf{O} \text{ wins the game } \mathbf{1} \vdash \mathcal{A}(\dot{T})$$

where, using the notions of App. A, an O-strategy is just a P-strategy on the dual game.

For ($\Rightarrow$), assuming given a winning P-strategy σ on $\mathcal{A}^{\perp\!\!\!\perp}(\dot{T})$, we build a winning O-strategy τ in $\mathcal{A}(\dot{T})$. The strategy τ is built by induction on plays. To each play t of τ, we associate a play s of σ such that if t leads to state $q_{\mathcal{A}}$, then s leads to state $(q_{\mathcal{A}}, \mathbb{f})$. In the base case, both t and s are the empty plays, and the invariant is respected. For the induction step, assume that P plays u from t in $\mathcal{A}(\dot{T})$. Let (f,F) be the move of σ from s. We then let τ answer the pair $(F(u,f(u)), f(u))$ from $s.u$, and $\mathcal{A}$ goes to state $q'_{\mathcal{A}}$. In $\mathcal{A}^{\perp\!\!\!\perp}(\dot{T})$, we let O play the pair $(f(u),u)$. Then $\mathcal{A}^{\perp\!\!\!\perp}$ goes to state $(q'_{\mathcal{A}}, \mathbb{f})$ and the invariant is respected. Since σ is winning and $\mathcal{A}^{\perp\!\!\!\perp}$ stays in states of the form $(_, \mathbb{f})$ the infinite sequence of states produced in $\mathcal{A}(\dot{T})$ is rejecting, as required.

For the converse direction ($\Leftarrow$), assuming given a winning O-strategy τ on $\mathcal{A}(\dot{T})$, we build a winning P-strategy σ in $\mathcal{A}^{\perp\!\!\!\perp}(\dot{T})$. The strategy σ is built by induction on plays as long as $\mathcal{A}^{\perp\!\!\!\perp}$ stays in states of the form $(_, \mathbb{f})$ (if it switches to $(_, \mathbb{t})$ then P trivially wins). So to each play s of σ which leads to state $(q_{\mathcal{A}}, \mathbb{f})$, we associate a play t of τ which leads to state $q_{\mathcal{A}}$. The base case is trivial. For the induction step, we build (f,F) from σ as follows: to each u, σ associates (from t) a pair (x,d). We let $F(u,_) := d$ and $f(u) := x$. Assume then that from $s.(f,F)$, O plays some (u,d). If $d \neq f(u)$ then we are done. Otherwise, $\mathcal{A}^{\perp\!\!\!\perp}$ switches to $(q'_{\mathcal{A}}, \mathbb{f})$. We then let P play u from t, so that by construction τ answers $(F(u,_), d)$, and $\mathcal{A}$ goes to state $q'_{\mathcal{A}}$. But then, since τ is winning for O, the sequence of $\mathcal{A}$-states is rejecting, so that P wins in $\mathcal{A}^{\perp\!\!\!\perp}(\dot{T})$, as required. □

5.4.1. Deduction Rules for $\perp\!\!\!\perp$ and $\mathcal{A}^{\perp\!\!\!\perp}$. Since the fibre categories $\mathsf{DialAut}_\Sigma$ are symmetric monoidal closed, they are in particular dialogue categories in the sense of [Mel13], with as exponentiating object any object of $\mathsf{DialAut}_\Sigma$. Hence, if as in Ex. 5.9 we extend the deduction system of Fig. 5.9 with the realizable rules

$$\overline{\mathcal{A} \multimap \perp\!\!\!\perp \vdash \mathcal{A}^{\perp\!\!\!\perp}} \qquad \text{and} \qquad \overline{\mathcal{A}^{\perp\!\!\!\perp} \vdash \mathcal{A} \multimap \perp\!\!\!\perp}$$

then we can derive the following rules for $\perp\!\!\!\perp$ and $\mathcal{A}^{\perp\!\!\!\perp}$:

$$\frac{\mathcal{A},\, \mathcal{B} \vdash \perp\!\!\!\perp}{\mathcal{A} \vdash \mathcal{B}^{\perp\!\!\!\perp}} \qquad \frac{\mathcal{A} \vdash \mathcal{B}^{\perp\!\!\!\perp}}{\mathcal{A},\, \mathcal{B} \vdash \perp\!\!\!\perp} \qquad \frac{\mathcal{A} \vdash \mathcal{B}^{\perp\!\!\!\perp}}{\mathcal{B} \vdash \mathcal{A}^{\perp\!\!\!\perp}} \qquad \overline{\mathcal{A} \vdash \mathcal{A}^{\perp\!\!\!\perp\perp\!\!\!\perp}} \qquad \frac{\mathcal{A} \vdash \mathcal{B}}{\mathcal{B}^{\perp\!\!\!\perp} \vdash \mathcal{A}^{\perp\!\!\!\perp}} \qquad \overline{\mathcal{A}^{\perp\!\!\!\perp\perp\!\!\!\perp\perp\!\!\!\perp} \vdash \mathcal{A}^{\perp\!\!\!\perp}}$$

6. Quantifications

We now discuss quantifications in the fibration $\mathsf{DialAut}$. We follow the categorical approach outlined in §2.2, according to which existential and universal quantifications (also called simple coproducts and products [Jac01, Chap. 1]) in a fibration $\mathsf{p} : \mathbb{E} \to \mathbb{B}$ are given resp. by left adjoints $\coprod_{I,J} : \mathbb{E}_{I\times J} \to \mathbb{E}_I$ and right adjoints $\prod_{I,J} : \mathbb{E}_{I\times J} \to \mathbb{E}_I$ to the *weakening functors* $\pi^\star : \mathbb{E}_I \to \mathbb{E}_{I\times J}$ induced by $\mathbb{B}$-projections $\pi : I \times J \to I$. The families of operations $(\coprod_{I,J})_{I,J}$ and $(\prod_{I,J})_{I,J}$ are moreover required to satisfy some coherence conditions, called the *Beck-Chevalley* conditions, which insure that they are preserved by substitution.

Having both (categorical) existential and universal quantifications greatly simplifies some basic reasoning on games (see Cor. 6.5 and Ex. 6.10). Referring to Rem. 3.11, this also allows for a clearer connection with Gödel's *Dialectica* interpretation (Ex. 6.4).

We first present quantifications in $\mathsf{DialAut}$ (§6.1), from which we then derive quantifications on automata (§6.2) and deduction rules for quantifications (§6.3).

6.1. Quantifications in $\mathsf{DialAut}$

Quantifications in $\mathsf{DialAut}$ are induced by quantifications in DialZ, which are themselves based on quantifications in simple fibrations. It is well-known (see e.g. [Jac01, Chap. 1]) that the simple fibration $\mathsf{s} : \mathsf{s}(\mathbb{B}) \to \mathbb{B}$ always has simple coproducts, and has simple products iff $\mathbb{B}$ is Cartesian closed. They are given by

$$\coprod_{I,J}(I \times J, X) \quad := \quad (I, J \times X) \qquad \text{and} \qquad \prod_{I,J}(I \times J, X) \quad := \quad (I, X^J)$$

This directly extends to DialZ.

Proposition 6.1. *The weakening functors* $[\pi]^\star : \mathsf{DialZ}(\Sigma) \to \mathsf{DialZ}(\Sigma\times\Gamma)$ *induced by projections* $\pi : \Sigma \times \Gamma \to \Sigma$ *have left and right adjoints given by*

$$\coprod_{\Sigma,\Gamma}(U, X) := (\Gamma \times U, X) \qquad \textit{and} \qquad \prod_{\Sigma,\Gamma}(U, X) := (U^\Gamma, \Gamma \times X) \simeq (\Gamma \multimap_{\mathbf{DZ}_{\mathfrak{D}}} (U, X))$$

Proof. Fix Σ, Γ and a projection $\pi : \Sigma \times \Gamma \to \Sigma$. According to [ML98, Thm. IV.1.2], we have to show that for each $\mathsf{DialZ}(\Sigma \times \Gamma)$-object A, there are $\mathsf{DialZ}(\Sigma \times \Gamma)$-morphisms

$$\eta_A \ : \ A \ \multimap_{\mathsf{DialZ}(\Sigma\times\Gamma)} \ [\pi]^\star(\coprod_{\Sigma,\Gamma} A) \qquad \text{and} \qquad \varepsilon_A \ : \ [\pi]^\star(\prod_{\Sigma,\Gamma} A) \ \multimap_{\mathsf{DialZ}(\Sigma\times\Gamma)} \ A$$

satisfying the following universal properties: for each $\mathsf{DialZ}(\Sigma)$-object B and each $\mathsf{DialZ}(\Sigma \times \Gamma)$-morphisms

$$\sigma \ : \ A \ \multimap_{\mathsf{DialZ}(\Sigma\times\Gamma)} \ [\pi]^\star(B) \qquad \text{and} \qquad \varsigma \ : \ [\pi]^\star(B) \ \multimap_{\mathsf{DialZ}(\Sigma\times\Gamma)} \ A$$

there are unique $\mathsf{DialZ}(\Sigma)$-morphisms

$$\theta \ : \ \coprod_{\Sigma,\Gamma} A \ \multimap_{\mathsf{DialZ}(\Sigma)} \ B \qquad \text{and} \qquad \vartheta \ : \ B \ \multimap_{\mathsf{DialZ}(\Sigma)} \ \prod_{\Sigma,\Gamma} A$$

	$((\Sigma \times \Gamma) \times U\ ,\ X)$	$\xrightarrow{\tilde{\eta}_A}{}_{\circ\mathbf{DZ}_{\mathfrak{D}}}$	$(\Gamma \times U\ ,\ X)$	
	$\vdots$		$\vdots$	
O	$((\mathtt{a}, \mathtt{b}), u)$			
			$(\mathtt{b}, u)$	P
			(x, d)	O
P	x			
	$\vdots$		$\vdots$	

	$((\Sigma \times \Gamma) \times U^{\Gamma}\ ,\ \Gamma \times X)$	$\xrightarrow{\tilde{\varepsilon}_A}{}_{\circ\mathbf{DZ}_{\mathfrak{D}}}$	$(U\ ,\ X)$	
	$\vdots$		$\vdots$	
O	$((\mathtt{a}, \mathtt{b}), f)$			
			$f(\mathtt{b})$	P
			(x, d)	O
P	$(\mathtt{b}, x)$			
	$\vdots$		$\vdots$	

Figure 19: The $\mathbf{DZ}_{\mathfrak{D}}$-morphisms $\tilde{\eta}_A$ and $\tilde{\varepsilon}_A$ in the proof of Prop. 6.1

such that we have

$$\begin{tikzcd} A \arrow[r, "\eta_A"] \arrow[d, "\sigma"'] & {[\pi]^\star(\coprod_{\Sigma,\Gamma} A)} \arrow[dl, "{[\pi]^\star(\theta)}"] \\ {[\pi]^\star(B)} & \end{tikzcd} \qquad \begin{tikzcd} {[\pi]^\star(B)} \arrow[d, "{[\pi]^\star(\vartheta)}"'] \arrow[dr, "\varsigma"] & \\ {[\pi]^\star(\coprod_{\Sigma,\Gamma} A)} \arrow[r, "\varepsilon_A"'] & A \end{tikzcd} \tag{33}$$

Now, since $[\pi]^\star$ is the identity on objects, writing $A = (U, X)$, the maps η_A and ε_A actually have type:

$$\begin{array}{lcccl} & \eta_A & : & (U\ ,\ X) & \longrightarrow_{\mathsf{DialZ}(\Sigma\times\Gamma)} & ((\Gamma \times U)\ ,\ X) \\ \text{and} & \varepsilon_A & : & (U^{\Gamma}\ ,\ \Gamma \times X) & \longrightarrow_{\mathsf{DialZ}(\Sigma\times\Gamma)} & (U\ ,\ X) \end{array}$$

They are induced from the $\mathbf{DZ}_{\mathfrak{D}}$-morphisms

$$\begin{array}{lcccl} & \tilde{\eta}_A & : & ((\Sigma \times \Gamma) \times U\ ,\ X) & \multimap_{\mathbf{DZ}_{\mathfrak{D}}} & (\Gamma \times U\ ,\ X) \\ \text{and} & \tilde{\varepsilon}_A & : & ((\Sigma \times \Gamma) \times U^{\Gamma}\ ,\ \Gamma \times X) & \multimap_{\mathbf{DZ}_{\mathfrak{D}}} & (U\ ,\ X) \end{array}$$

depicted on Fig. 19 and themselves based on the monoidal closed structure of $\mathbf{DZ}_{\mathfrak{D}}$.

The existence and uniqueness of θ and ϑ satisfying (33) follow from the fact that comonoids have a Cartesian structure and from the monoidal closure of $\mathbf{DZ}_{\mathfrak{D}}$. □

The Beck-Chevalley conditions amount, for $L \in \mathbf{T}[\Delta, \Sigma]$, to the equalities

$$L^\star(\underset{\Sigma,\Gamma}{\square}(U, X)) \quad = \quad \underset{\Delta,\Gamma}{\square}(L \times \mathrm{Id}_\Gamma)^\star(U, X) \qquad \text{for } \square \in \{\coprod, \prod\}$$

which follow from the fact that substitution functors are identities on objects.

The extension to $\mathsf{DialAut}$ just requires to handle winning and acceptance.

Proposition 6.2. *The fibration* DialAut *has existential and universal quantifications given by*

$$\coprod_{\Sigma,\Gamma}(U,X,\mathcal{W}_A) \;:=\; (\Gamma\times U, X, \coprod_{\Sigma,\Gamma}\mathcal{W}_A) \quad \textit{and} \quad \prod_{\Sigma,\Gamma}(U,X,\mathcal{W}_A) \;:=\; (U^\Gamma, \Gamma\times X, \prod_{\Sigma,\Gamma}\mathcal{W}_A)$$

where $\coprod \mathcal{W}_A$ *is defined from* $\mathcal{W}_A$ *via associativity and* $\prod \mathcal{W}_A$ *by pointwise function application as* $((\mathsf{a}_k, f_k)\cdot(\mathsf{b}_k, x_k, d_k))_k \in \prod_{\Sigma,\Gamma}\mathcal{W}_A$ *iff* $((\mathsf{a}_k, \mathsf{b}_k, f_k(\mathsf{b}_k))\cdot(x_k,d_k))_k \in \mathcal{W}_A$.

Proof. It is easy to check that for $A = (U,X,\mathcal{W}_A)$ over $\Sigma\times\Gamma$, the universal $\mathsf{DialZ}(\Sigma\times\Gamma)$-morphisms $\eta_{(U,X)}$ and $\varepsilon_{(U,X)}$ induce $\mathsf{DialAut}_{\Sigma\times\Gamma}$-morphisms, that is, according to Def. 4.7, that their lifts $\iota(\eta^{\uparrow}_{(U,X)})$ and $\iota(\varepsilon^{\uparrow}_{(U,X)})$ are winning.

The Beck-Chevalley conditions, which amount to

$$L^\star(\coprod_{\Sigma,\Gamma} A) = \coprod_{\Delta,\Gamma}(L\times \mathrm{Id}_\Gamma)^\star(A) \qquad \text{and} \qquad L^\star(\prod_{\Sigma,\Gamma} A) = \prod_{\Delta,\Gamma}(L\times \mathrm{Id}_\Gamma)^\star(A)$$

are straightforward from the definitions. □

6.2. Quantifications on Uniform Automata

Similarly as with the monoidal closed structure, the quantifications on automata and their deduction rules are obtained by direct adaptation of the quantifications of DialAut.

Definition 6.3. *Given* $\mathcal{A}:\Sigma\times\Gamma$ *with set of* P*-moves* U *and set of* O*-moves* X*, let*

$$\begin{array}{lcl}(\exists_\Gamma\mathcal{A}:\Sigma) & := & (Q_\mathcal{A}\,,\, q^i_\mathcal{A}\,,\, \Gamma\times U\,,\, X\,,\, \partial_{\exists_\Gamma\mathcal{A}}\,,\, \Omega_\mathcal{A})\\ (\forall_\Gamma\mathcal{A}:\Sigma) & := & (Q_\mathcal{A}\,,\, q^i_\mathcal{A}\,,\, U^\Gamma\,,\, \Gamma\times X\,,\, \partial_{\forall_\Gamma\mathcal{A}}\,,\, \Omega_\mathcal{A})\end{array}$$

where

$$\begin{array}{lcl}\partial_{\exists_\Gamma\mathcal{A}}(q,\,\mathsf{a},\,(\mathsf{b},u),\,x,\,d) & := & \partial_\mathcal{A}(q,\,(\mathsf{a},\mathsf{b}),\,u,\,x,\,d)\\ \textit{and}\qquad \partial_{\forall_\Gamma\mathcal{A}}(q,\,\mathsf{a},\,f,\,(\mathsf{b},x),\,d) & := & \partial_\mathcal{A}(q,\,(\mathsf{a},\mathsf{b}),\,f(\mathsf{b}),\,x,\,d)\end{array}$$

Quantifications on automata induce an $\exists\forall$-structure which is reminiscent from Gödel's *Dialectica* interpretation (see e.g. [AF98, Koh08]).

Example 6.4. *Given* $\mathcal{A}:\Sigma$ *with set of* P*-moves* U *and set of* O*-moves* X*, let* $\mathcal{D}$ *be the deterministic automaton*

$$(\mathcal{D}:\Sigma\times U\times X) \quad := \quad (Q_\mathcal{A},\, q^i_\mathcal{A},\, \mathbf{1},\, \mathbf{1},\, \partial_\mathcal{D},\, \Omega_\mathcal{A})$$

whose transition function

$$\partial_\mathcal{D} \quad : \quad Q_\mathcal{A}\times(\Sigma\times U\times X) \quad\longrightarrow\quad \mathfrak{D} \quad\longrightarrow\quad Q_\mathcal{A}$$

is obtained from $\partial_\mathcal{A}$ *in the obvious way. In* $\mathsf{DialAut}_\Sigma$ *we have* $\mathcal{A}\simeq\exists_U\forall_X\mathcal{D}$.

Let us now discuss the connection between quantifications on automata and in DialAut. First, given $(\mathcal{A}:\Sigma\times\Gamma)$, we have, as $\mathsf{DialAut}_\Sigma$-objects,

$$(\Sigma\vdash\coprod_{\Sigma,\Gamma}\mathcal{A}) \quad = \quad (\Sigma\vdash\exists_\Sigma\mathcal{A}) \qquad \text{and} \qquad (\Sigma\vdash\prod_{\Sigma,\Gamma}\mathcal{A}) \quad = \quad (\Sigma\vdash\forall_\Sigma\mathcal{A})$$

It then follows that the Beck-Chevalley conditions in DialAut imply

$$\begin{array}{lcl}\coprod_{\Sigma,\Gamma}\mathcal{A}(M\times\mathrm{Id}_\Gamma) & = & M^\star(\coprod_{\Delta,\Gamma}\mathcal{A}) \;=\; (\exists_\Gamma\mathcal{A})(M)\\ \prod_{\Sigma,\Gamma}\mathcal{A}(M\times\mathrm{Id}_\Gamma) & = & M^\star(\prod_{\Delta,\Gamma}\mathcal{A}) \;=\; (\forall_\Gamma\mathcal{A})(M)\end{array}$$

Thanks to the adjunctions $\coprod \dashv \pi^\star \dashv \prod$ in $\mathsf{DialAut}$, we then have

$$\begin{array}{lcl} \Sigma \vdash (\exists_\Gamma \mathcal{A})(M) \multimap \mathcal{B}(N) & \simeq & \Sigma \times \Gamma \vdash \mathcal{A}(M \times \mathrm{Id}_\Gamma) \multimap \mathcal{B}(N \circ [\pi_\Sigma]) \\ \Sigma \vdash \mathcal{B}(N) \multimap (\forall_\Gamma \mathcal{A})(M) & \simeq & \Sigma \times \Gamma \vdash \mathcal{B}(N \circ [\pi_\Sigma]) \multimap \mathcal{A}(M \times \mathrm{Id}_\Gamma) \end{array} \tag{34}$$

It follows that P has winning strategies in

$$\Sigma \times \Gamma \vdash (\forall_\Gamma \mathcal{A})[\pi_\Sigma] \longrightarrow \mathcal{A} \qquad \text{and} \qquad \Sigma \times \Gamma \vdash \mathcal{A} \longrightarrow (\exists_\Gamma \mathcal{A})[\pi_\Sigma] \tag{35}$$

We thus get the following corollary to Prop. 6.2.

Corollary 6.5. *Given uniform automata $\mathcal{A}, \mathcal{B} : \Sigma$, the game $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$ is equivalent to a regular game on a finite graph. It is therefore decidable whether there exists a winning P-strategy on $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$, and if there exists such a winning P-strategy, then there exists a finite-state one, which is moreover effectively computable from $\mathcal{A}$ and $\mathcal{B}$.*

Proof. By (34) and (35), P has a winning strategy in $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$ iff it has a winning strategy in $\mathbf{1} \vdash \mathbf{I_1} \multimap \forall_\Sigma(\mathcal{A} \multimap \mathcal{B})$. But since in that game O can only play $\bullet$ in the component $\mathbf{I_1}$, similarly as in §3.5, it is equivalent to the acceptance game of the automaton $\forall_\Sigma(\mathcal{A} \multimap \mathcal{B}) : \mathbf{1}$ on the unique tree $\mathbf{1} : \mathfrak{D}^* \to \mathbf{1}$.

Reasoning as in [Tho97, Ex. 6.12], the game $\mathbf{1} \vdash \forall_\Sigma(\mathcal{A} \multimap \mathcal{B})$ is effectively equivalent to a regular game on a finite graph. Then, by the Büchi-Landweber Theorem [BL69] (see also [Tho97, Thm. 6.18]), one can decide which player has a winning strategy, and the winner always has a finite-state winning strategy which is moreover effectively computable from the game graph. □

We also get from (35) that existential quantifications are complete in the following sense:

Corollary 6.6. *Given $\mathcal{A} : \Sigma \times \Gamma$, we have $\pi_\Gamma(\mathcal{L}(\mathcal{A})) \subseteq \mathcal{L}(\exists_\Gamma \mathcal{A})$.*

The converse inclusion (the correctness of existential quantifications) only holds for *non-deterministic* automata, and is detailed in §7. Dually, it follows from (35) that universal quantifications are correct (but they are complete only on *universal* automata, see Def. 3.1).

Corollary 6.7. *Given $\mathcal{A} : \Sigma \times \Gamma$, if $T \in \mathcal{L}(\forall_\Gamma \mathcal{A})$, then for all Γ-labeled tree T' we have $\langle T, T' \rangle \in \mathcal{L}(\mathcal{A})$.*

6.3. Deduction Rules for Quantifications

We now turn to deduction rules for quantification. It follows from the isos (34) that we can extend the deduction system of Fig. 18 with the rules of Fig. 20 while preserving adequacy (Prop. 5.6), Ex. 5.9 and compatibility with cut-elimination (in the sense of Rem. 5.7).

Proposition 6.8 (Adequacy with Quantifications)**.** *If the sequent $M \;;\; \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B}$ is derivable using the rules of Fig. 18, Fig. 20 and of Ex. 5.9, then there is a winning P-strategy in the game*

$$\mathcal{A}_1(M) \otimes_{\mathsf{DA}} \cdots \otimes_{\mathsf{DA}} \mathcal{A}_n(M) \quad \longrightarrow \quad \mathcal{B}(M)$$

Note that the rules of Fig. 20 involve internalized substitutions of the form $\mathcal{A}[\mathtt{f}]$ as defined in Def. 3.7. The transfer rules ($\textsc{Trans}_\uparrow$) and ($\textsc{Trans}_\downarrow$) allow to connect the internalized substitutions of the form $\mathcal{A}[\mathtt{f}]$ with the $\mathbf{T}$-substitution.

$$(\textsc{Subst})\quad \frac{M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A}}{M \circ M'\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A}}$$

$$(\textsc{Trans}_{\downarrow})\quad \frac{[\mathtt{f}] \circ M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{B}}{M\ ;\ \overline{\mathcal{A}[\mathtt{f}]} \vdash \mathcal{B}[\mathtt{f}]} \qquad \frac{M\ ;\ \overline{\mathcal{A}[\mathtt{f}]} \vdash \mathcal{B}[\mathtt{f}]}{[\mathtt{f}] \circ M\ ;\ \overline{\mathcal{A}} \vdash \mathcal{B}}\quad (\textsc{Trans}_{\uparrow})$$

$$(\textsc{Left}\ \exists)\quad \frac{M \times \mathrm{Id}_\Gamma\ ;\ \overline{\mathcal{A}[\pi]}, \mathcal{B} \vdash \mathcal{A}[\pi]}{M\ ;\ \overline{\mathcal{A}}, \exists_\Gamma \mathcal{B} \vdash \mathcal{A}} \qquad \frac{M \times N\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A}}{M \times N\ ;\ \overline{\mathcal{A}} \vdash (\exists_\Gamma \mathcal{A})[\pi]}\quad (\textsc{Right}\ \exists)$$

$$(\textsc{Left}\ \forall)\quad \frac{M \times N\ ;\ \overline{\mathcal{A}}, \mathcal{B} \vdash \mathcal{A}}{M \times N\ ;\ \overline{\mathcal{A}}, (\forall_\Gamma \mathcal{B})[\pi] \vdash \mathcal{A}} \qquad \frac{M \times \mathrm{Id}_\Gamma\ ;\ \overline{\mathcal{A}[\pi]} \vdash \mathcal{A}}{M\ ;\ \overline{\mathcal{A}} \vdash \forall_\Gamma A}\quad (\textsc{Right}\ \forall)$$

Figure 20: Substitution and quantification rules for uniform automata (where M, M' are composable, π is a suitable projection and $\mathtt{f}$ is a function on alphabets)

Example 6.9. *Using the transfer rule* ($\textsc{Trans}_{\downarrow}$)*, we can derive the following specific rules of substitution for* $\mathbf{T}$*-maps induced by functions* $\mathtt{f} : \Sigma \to \Gamma$*:*

$$\frac{\mathrm{Id}_\Gamma\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A}}{\mathrm{Id}_\Sigma\ ;\ \overline{\mathcal{A}[\mathtt{f}]} \vdash \mathcal{A}[\mathtt{f}]} \qquad \frac{M \times \mathrm{Id}_\Gamma\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A}}{M \times \mathrm{Id}_\Sigma\ ;\ \overline{\mathcal{A}[\mathrm{id} \times \mathtt{f}]} \vdash \mathcal{A}[\mathrm{id} \times \mathtt{f}]}$$

Indeed, since we have (as $\mathbf{T}$*-morphisms)*

$$\mathrm{Id}_\Gamma \circ [\mathtt{f}] \quad = \quad [\mathtt{f}] \circ \mathrm{Id}_\Sigma \qquad \textit{and} \qquad (M \times \mathrm{Id}_\Gamma) \circ [\mathrm{id} \times \mathtt{f}] \quad = \quad (\mathrm{id} \times \mathtt{f}) \circ (M \times \mathrm{Id}_\Sigma)$$

it follows that we can derive

$$\cfrac{\cfrac{\mathrm{Id}_\Gamma\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A}}{\mathrm{Id}_\Gamma \circ [\mathtt{f}]\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A}}}{\mathrm{Id}_\Sigma\ ;\ \overline{\mathcal{A}[\mathtt{f}]} \vdash \mathcal{A}[\mathtt{f}]} \qquad \textit{and} \qquad \cfrac{\cfrac{M \times \mathrm{Id}_\Gamma\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A}}{(M \times \mathrm{Id}_\Gamma) \circ [\mathrm{id} \times \mathtt{f}]\ ;\ \overline{\mathcal{A}} \vdash \mathcal{A}}}{M \times \mathrm{Id}_\Sigma\ ;\ \overline{\mathcal{A}[\mathrm{id} \times \mathtt{f}]} \vdash \mathcal{A}[\mathrm{id} \times \mathtt{f}]}$$

Example 6.10. *Continuing Ex. 5.9, we can extend the deduction system with the rule*

$$\frac{\mathcal{L}(\mathcal{A} : \mathbf{1}) \neq \emptyset}{\vdash \mathcal{A}}$$

This rule actually subsumes Ex. 5.9. Indeed, following the same reasoning as for Cor. 6.5, assuming that

$$\Sigma\ ;\ \mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \vdash \mathcal{B}$$

is realizable we get (leaving implicit some structural and cut rules)

$$\cfrac{\cfrac{\cfrac{\cfrac{\mathcal{L}(\forall_\Sigma(\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B})) \neq \emptyset}{\mathbf{1}\ ;\ \vdash \forall_\Sigma(\mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B})}}{\Sigma\ ;\ \vdash \mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \multimap \mathcal{B}}}{\Sigma\ ;\ \mathcal{A}_1 \otimes \cdots \otimes \mathcal{A}_n \vdash \mathcal{B}}}{\Sigma\ ;\ \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B}}$$

7. Non-Deterministic Automata

This final Section focuses on structural properties of non-deterministic automata, on their role in Rabin's Theorem [Rab69] (namely in the complementation of non-deterministic tree automata), and on their relationship with IMELL [Gir87] (see §1.2, §1.3 and §2.5).

We first detail in §7.1 the Cartesian structure of non-deterministic automata announced in §1.3 (see also §2.4 and §5.3). Technically, this Cartesian structure follows from the simple fact that non-deterministic automata generate comonoids in the fibres of DialAut (by a direct extension of Prop. 4.4, §4.3). As a consequence, we show that our model has the witnessing properties asked to computational interpretations of proofs (in the sense of §1.4), and moreover that it allows for combining strategies obtained from proofs with witnessing strategies computed by usual emptiness checking algorithms (see §1.5).

Second, we show that a powerset construction for the *Simulation Theorem* [MS87, EJ91, MS95] satisfies the usual deduction rules of the exponential modality ! of IMELL. This completes the picture sketched in §1.3, §1.5, §2.4 and §2.5, and moreover allows us to obtain a deduction system which is complete w.r.t. intuitionistic and classical deduction (via usual translations). Furthermore, App. C details how two constructions from resp. [CL08] and [SA05] can be reformulated in our setting.

7.1. The Cartesian Structure of Non-Deterministic Automata

Similarly as with usual (total) non-deterministic automata (see §2.4), the monoidal product of uniform automata is Cartesian on non-deterministic automata. Recall from Def. 3.1 that a uniform automaton is non-deterministic if its set of O-moves is $\simeq \mathbf{1}$.

Consider a $\mathsf{DialAut}_\Sigma$-object $\mathcal{N}(L)$ with $\mathcal{N}$ non-deterministic and with set of P-moves U. Hence, the underlying $\mathsf{DialZ}(\Sigma)$-object of $\mathcal{N}(L)$ is of the form (U, I) with $I \simeq \mathbf{1}$. As we have seen in §5.3, we thus get canonical realizers for

$$\mathcal{N}(L) \quad \multimap \quad \mathcal{N}(L) \otimes \mathcal{N}(L) \qquad \text{and} \qquad \mathcal{N}(L) \quad \multimap \quad \mathbf{I} \tag{36}$$

As we shall see now, these canonical realizer equip $\mathcal{N}(L)$ with the structure of a comonoid[15]. Thanks to well-known results (see e.g. [Mel09, Cor. 18, §6.5]), this implies that the monoidal structure of uniform automata is Cartesian on non-deterministic automata.

Recall from Prop. 4.2 that objects of the form (K, I) with $I \simeq \mathbf{1}$ are comonoids in $\mathbf{DZ}$, and from Prop. 4.4 that such objects are also comonoids in $\mathbf{DZ}_{\mathfrak{D}}$. On the other hand, we have seen that $\mathsf{DialZ}(\Sigma)$ is a Kleisli category of comonoid indexing in $\mathbf{DZ}_{\mathfrak{D}}$, whose symmetric monoidal structure is given by the extension of Prop. 4.4 to comonoid indexing given by Prop 5.2. Actually, the lifting of comonoids given by Prop. 4.4 also extends to the case of comonoid indexing:

Proposition 7.1. *Given a comonoid C in a symmetric monoidal category $(\mathbb{C}, \otimes, \mathbf{I})$, each comonoid (K, d, e) in $\mathbb{C}$ induces a comonoid $(K,\ d \circ \varepsilon_K^C,\ e \circ \varepsilon_K^C)$ in the Kleisli category $\mathbf{Kl}(C)$ of indexing with C. In the case of $\mathsf{DialZ}(\Sigma)$, the structure maps $\tilde{d}_K$ and $\tilde{e}_K$ of the comonoid induced by $K = (K, \mathbf{1})$ can be depicted as on Fig. 22 (where we omitted some $\bullet$-moves).*

The extension of Prop. 7.1 to the $\mathsf{DialAut}_\Sigma$-objects induced by non-deterministic automata is direct. Moreover, $\mathsf{DialAut}_\Sigma$-morphisms between non-deterministic automata are comonoid morphisms.

[15] Recall from §4.2 that in this paper, by (co)monoid we always mean *commutative* (co)monoid

Proposition 7.2. *For each alphabet Σ, objects of the form $\Sigma \vdash \mathcal{N}(L)$, where $\mathcal{N}$ is non-deterministic, are comonoids in* $\mathsf{DialAut}_\Sigma$. *Moreover,* $\mathsf{DialAut}_\Sigma$*-morphisms between such objects are comonoid morphisms.*

Proof. Consider first a $\mathsf{DialAut}_\Sigma$-object $\mathcal{N}(L)$ with $\mathcal{N}$ non-deterministic and with set of P-moves U and set of O-moves $I \simeq \mathbf{1}$. Hence (U, I) is a comonoid in $\mathsf{DialZ}(\Sigma)$ by Prop. 7.1. Moreover, the comonoid structure maps play as the maps depicted on Fig. 22 (replacing $(K, \mathbf{1})$ with (U, I)), and winning is trivial.

Consider now a $\mathsf{DialAut}_\Sigma$-morphism $\sigma : \mathcal{N}(L) \multimap \mathcal{K}(L')$, where $\mathcal{N}$ and $\mathcal{K}$ are non-deterministic, with sets of P-moves resp. U and V, and sets of O-moves resp. I and J (where $I \simeq J \simeq \mathbf{1}$). We show that σ is a comonoid map by reasoning similarly as in Prop. 4.5 for the base category $\mathbf{T}$. Writing $\tilde{d}_\mathcal{N}$ and $\tilde{e}_\mathcal{N}$ (resp. $\tilde{d}_\mathcal{K}$ and $\tilde{e}_\mathcal{K}$) for the comonoid structure maps of $\mathcal{N}(L)$ (resp. $\mathcal{K}(L')$), we have to show that the following equations hold in $\mathsf{DialAut}_\Sigma$:

$$(\sigma \otimes \sigma) \circ \tilde{d}_\mathcal{N} \quad = \quad \tilde{d}_\mathcal{K} \circ \sigma \qquad \text{and} \qquad \tilde{e}_\mathcal{N} \quad = \quad \tilde{e}_\mathcal{K} \circ \sigma$$

Assuming that σ plays as in Fig. 21 (top left). The first equation follows from the fact that $\tilde{d}_\mathcal{K} \circ \sigma$ plays as in Fig. 21 (middle), while $(\sigma \otimes \sigma) \circ \tilde{d}_\mathcal{M}$ plays as in Fig. 21 (bottom). The second equation follows from the fact that $\tilde{e}_\mathcal{K} \circ \sigma$ plays as in Fig. 21 (top right). □

Since the category of comonoids of a symmetric monoidal category has finite products (see e.g. [Mel09, Cor. 18, §6.5]), we thus have the expected result that non-deterministic automata are equipped with a Cartesian structure.

Corollary 7.3. *For each alphabet Σ, the full subcategory* $\mathsf{DialAut}_\Sigma^{\mathrm{ND}}$ *of* $\mathsf{DialAut}_\Sigma$*, whose objects are of the form $(U, I, \mathcal{W})$ with $I \simeq \mathbf{1}$, is Cartesian.*

7.1.1. Application: Deduction Rules for Non-Deterministic Automata. Similarly as with usual (total) non-deterministic automata (see §2.4), Cor. 7.3 allows us to extend adequacy (Prop. 5.6 and Prop. 6.8) to the following restriction of the structural weakening and contraction rules:

$$(\textsc{Weak}_{\mathrm{ND}}) \quad \frac{M \;;\; \overline{\mathcal{A}}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M \;;\; \overline{\mathcal{A}}, \mathcal{N}, \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad\qquad \frac{M \;;\; \overline{\mathcal{A}}, \mathcal{N}, \mathcal{N}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M \;;\; \overline{\mathcal{A}}, \mathcal{N}, \overline{\mathcal{B}} \vdash \mathcal{C}} \quad (\textsc{Contr}_{\mathrm{ND}}) \tag{37}$$

where $\mathcal{N}$ is required to be non-deterministic (while $\overline{\mathcal{A}}$, $\overline{\mathcal{B}}$ and $\mathcal{C}$ can be arbitrary). On the other hand, recall that the full weakening rule is actually derivable in the setting of Ex. 5.9, but with non-canonical realizers of $\mathcal{A} \multimap \mathbf{I}$ when $\mathcal{A}$ is not non-deterministic.

7.1.2. Application: Existential Quantifications and Extraction. A nice consequence of the Cartesian structure of $\mathsf{DialAut}_{(-)}^{\mathrm{ND}}$ is the fact that existential quantifications behave similarly as the usual *sum types* of Type Theory (see e.g. [Jac01, Chap. 10]). Consider a non-deterministic automaton $\mathcal{N} : \Sigma \times \Gamma$ with set of P-moves U, and let T be a Σ-labeled tree (so that $T : \mathfrak{D}^* \to \Sigma$). It directly follows from Def. 3.9 that a winning P-strategy in $\mathbf{1} \vdash \mathbf{I} \multimap (\exists_\Gamma \mathcal{A})(\dot{T})$ is given by a function

$$\bigcup_{n \in \mathbb{N}} \mathfrak{D}^n \quad \longrightarrow \quad \Gamma \times U$$

hence by a pair of functions

$$\left(\bigcup_{n \in \mathbb{N}} \mathfrak{D}^n \quad \longrightarrow \quad \Gamma \right) \times \left(\bigcup_{n \in \mathbb{N}} \mathfrak{D}^n \quad \longrightarrow \quad U \right)$$

	$\mathcal{N}(L)$ $\xrightarrow{\sigma}$	$\mathcal{K}(L')$
	$\vdots$	$\vdots$
O	(a, u)	
P		v
O		d
P	$\bullet$	
	$\vdots$	$\vdots$

	$\mathcal{N}(L)$ $\xrightarrow{\sigma}$	$\mathcal{K}(L')$	$\mathcal{K}(L')$ $\xrightarrow{\tilde{e}_{\mathcal{K}}}$	$\mathbf{I}$	
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	
O	(a, u)				
		v	(a, v)		
				$\bullet$	P
				d	O
		d	$\bullet$		
P	$\bullet$				
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	

	$\mathcal{N}(L)$ $\xrightarrow{\sigma}$	$\mathcal{K}(L')$	$\mathcal{K}(L')$ $\xrightarrow{\tilde{d}_{\mathcal{K}}}$	$\mathcal{K}(L') \otimes \mathcal{K}(L')$	
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	
O	(a, u)				
		v	(a, v)		
				(v, v)	P
				d	O
		d	$\bullet$		
P	$\bullet$				
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	

	$\mathcal{N}(L)$ $\xrightarrow{\tilde{d}_{\mathcal{N}}}$	$\mathcal{N}(L) \otimes \mathcal{N}(L)$	$\mathcal{N}(L) \otimes \mathcal{N}(L)$ $\xrightarrow{\sigma \otimes \sigma}$	$\mathcal{K}(L') \otimes \mathcal{K}(L')$	
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	
O	(a, u)				
		(u, u)	$(\mathsf{a}, (u, u))$		
				(v, v)	P
				d	O
		d	$\bullet$		
P	$\bullet$				
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	

Figure 21: $\mathsf{DialAut}_{\Sigma}$-maps on non-det. automata as comonoids maps in Prop. 7.2

	$\Sigma \otimes K$ $\xrightarrow{\tilde{d}_K}_{\mathbf{DZ}_{\mathfrak{D}}}$	$K \otimes K$	
	$\vdots$	$\vdots$	
O	(a, k)		
		(k,k)	P
		d	O
P	$\bullet$		
	$\vdots$	$\vdots$	

	$\Sigma \otimes K$ $\xrightarrow{\tilde{e}_K}_{\mathbf{DZ}_{\mathfrak{D}}}$	$\mathbf{I}$	
	$\vdots$	$\vdots$	
O	(a, k)		
		$\bullet$	P
		d	O
P	$\bullet$		
	$\vdots$	$\vdots$	

Figure 22: Structure maps in $\mathsf{DialZ}(\Sigma)$ for the comonoid $K = (K, \mathbf{1})$

and therefore by a tree $T' : \mathfrak{D}^* \to \Gamma$ together with a winning P-strategy in $\mathbf{1} \vdash \mathbf{I} \multimap \mathcal{A}\langle \dot{T}, \dot{T}' \rangle$. We thus have shown

Proposition 7.4. *Given a non-deterministic automaton $\mathcal{N} : \Sigma \times \Gamma$, a winning* P*-strategy $\sigma : \mathbf{1} \multimap \exists_\Sigma \mathcal{N}$ is of the form $\sigma = \langle T, \tau \rangle$ where T is a Σ-labeled tree and τ is a winning* P*-strategy in $\mathbf{1} \multimap \mathcal{N}(T)$ (so in particular $T \in \mathcal{L}(\mathcal{N})$).*

In particular, we get the following fact, which completes Cor. 6.6 and mirrors the well-known situation with usual non-deterministic automata.

Corollary 7.5. *If $\mathcal{N} : \Sigma \times \Gamma$ is* non-deterministic *then $\mathcal{L}(\exists_\Gamma \mathcal{N}) = \pi_\Gamma(\mathcal{L}(\mathcal{N}))$.*

Moreover, it follows from Prop. 7.4 that our computational interpretation makes it possible to effectively extract witnesses from (interpretations of) proofs, in the sense of §1.4 and §1.5. Let $\mathcal{N} : \Sigma$ be non-deterministic with set of P-moves U, and consider a derivation $\mathscr{D}$ of the sequent

$$\mathbf{1} \; ; \vdash \exists_\Sigma \mathcal{N}$$

using the rules of Fig. 18, Fig. 20, Ex. 5.9 and (37). Then adequacy (Prop. 5.6 and Prop. 6.8) gives a strategy

$$\sigma \quad : \quad \mathbf{I} \quad \multimap \quad \exists_\Sigma \mathcal{N}$$

(effectively computed by induction on $\mathscr{D}$), and which by Prop. 7.4 is of the form

$$\begin{array}{rrcl} & \langle T, \tau \rangle & : & \bigcup_{n \in \mathbb{N}} \mathfrak{D}^n \quad \longrightarrow \quad \Sigma \times U \\ \text{where} & \tau & : & \mathbf{I} \quad \multimap \quad \mathcal{N}(T) \end{array}$$

7.1.3. Application: Effective Realizers from Witnesses of Non-Emptiness. Similarly as with usual non-deterministic automata (see e.g. [Tho97]), thanks to the Büchi-Landweber Theorem [BL69], Cor. 7.5 implies the decidability of emptiness for non-deterministic automata as well as the *Rabin Basis Theorem* [Rab72], stating that if $\mathcal{L}(\mathcal{N}) \neq \emptyset$, then its contains a regular tree T and a finite state winning P-strategy on $\mathcal{N}(T)$ (both effectively definable from $\mathcal{N}$).

Corollary 7.6. *Given a non-deterministic automaton $\mathcal{N} : \Sigma$, one can decide whether $\mathcal{L}(\mathcal{N})$ is empty. Moreover, if $\mathcal{L}(\mathcal{N}) \neq \emptyset$ then one can effectively build from $\mathcal{N}$ a regular tree $T \in \mathcal{L}(\mathcal{N})$ together with a finite state winning* P*-strategy on $\mathbf{I} \multimap \mathcal{N}(T)$.*

Proof. It follows from Cor. 7.5 that $\mathcal{L}(\mathcal{N})$ is not empty iff the automaton $(\exists_\Sigma \mathcal{N}) : \mathbf{1}$ accepts the unique $\mathbf{1}$-labeled tree $\mathbf{1}$. We then proceed similarly as in the proof of for Cor. 6.5: reasoning as in [Tho97, Ex. 6.12], the game $\mathbf{1} \vdash \exists_\Sigma \mathcal{N}$ is effectively equivalent to a regular game on a finite graph. Then, by Büchi-Landweber Theorem [BL69] (see also [Tho97, Thm. 6.18]), one can decide which player has a winning strategy, and the winner always has a finite-state winning strategy which is moreover effectively computable from the game graph. Now, this strategy can be lifted to a finite state winning strategy on $\mathbf{1} \vdash \exists_\Sigma \mathcal{N}$, and we can then conclude thanks to Prop. 7.4. □

More generally, strategies witnessing (non-)emptiness obtained via Cor. 7.5 can be lifted to winning strategies in games of the form $\mathcal{A} \multimap \mathcal{C}$. Consider the case (mentioned in §1.5.(2)) of $\mathcal{C} = \mathcal{B}^\perp$ and with $\mathcal{A}, \mathcal{B} : \Sigma$ non-deterministic. If $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B}) = \emptyset$, then an O-strategy witnessing $\mathcal{L}(\mathcal{A} \otimes \mathcal{B}) = \emptyset$, which corresponds via Prop. 5.12[16] to a P-strategy witnessing $\mathbf{1} \in \mathcal{L}((\exists_\Sigma(\mathcal{A} \otimes \mathcal{B}))^\perp)$, can be lifted to a winning P-strategy in $\mathcal{A} \multimap \mathcal{B}^\perp$.

Proposition 7.7. *Given non-deterministic $\mathcal{A}, \mathcal{B} : \Sigma$, if $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B}) = \emptyset$, then there are winning P-strategies in $\mathcal{A} \otimes \mathcal{B} \multimap \perp$ and $\mathcal{A} \multimap \mathcal{B}^\perp$. Moreover, these P-strategies can be assumed to be finite state and can be effectively obtained from $\mathcal{A}$ and $\mathcal{B}$.*

Proof. Since $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B}) = \emptyset$, we have $\mathcal{L}(\mathcal{A} \otimes \mathcal{B}) = \emptyset$ by Prop. 5.10. Since $\mathcal{A}$ and $\mathcal{B}$ are non-deterministic, so is $\mathcal{A} \otimes \mathcal{B}$. It then follows from Cor. 7.5 that $\mathcal{L}(\exists_\Sigma(\mathcal{A} \otimes \mathcal{B})) = \emptyset$, hence, by Prop. 5.12 that the automaton $(\exists_\Sigma(\mathcal{A} \otimes \mathcal{B}))^\perp : \mathbf{1}$ accepts the unique tree $\mathbf{1} : \mathfrak{D}^* \to \mathbf{1}$. But winning P-strategies in $(\exists_\Sigma(\mathcal{A} \otimes \mathcal{B}))^\perp(\mathbf{1})$ can be lifted to winning P-strategies in

$$\mathbf{1} \quad \vdash \quad \mathbf{I_1}(\mathbf{1}) \quad \multimap \quad (\exists_\Sigma(\mathcal{A} \otimes \mathcal{B}))^\perp(\mathbf{1})$$

But note that since $(\exists_\Sigma(\mathcal{A} \otimes \mathcal{B}))^\perp : \mathbf{1}$, that game is actually the same as

$$\mathbf{1} \quad \vdash \quad \mathbf{I_1} \quad \multimap \quad (\exists_\Sigma(\mathcal{A} \otimes \mathcal{B}))^\perp$$

It then follows from monoidal closure (Prop. 5.4) that there is a winning P-strategy in the game

$$\mathbf{1} \quad \vdash \quad \exists_\Sigma(\mathcal{A} \otimes \mathcal{B}) \quad \multimap \quad \perp$$

and therefore by Prop. 6.2 (in the form of (34)) that there is a winning P-strategy on $\mathcal{A} \otimes \mathcal{B} \multimap \perp$ and therefore also in $\mathcal{A} \multimap \mathcal{B}^\perp$.

Moreover, it follows from Cor. 7.6 that there is a finite-state winning P-strategy in the game $\mathbf{1} \vdash (\exists_\Sigma(\mathcal{A} \otimes \mathcal{B}))^\perp(\mathbf{1})$ which is easily seen to be lifted to finite state P-strategies in $\mathcal{A} \otimes \mathcal{B} \multimap \perp$ and $\mathcal{A} \multimap \mathcal{B}^\perp$. □

Proposition 7.7, together with Ex. 5.9.(ii), implies the following extension of Ex. 5.9.(i).

Corollary 7.8. *If $\mathcal{A}, \mathcal{B} : \Sigma$ are non-deterministic and such that $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B}) = \emptyset$, then $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B} \multimap \mathcal{A}) \subseteq \mathcal{L}(\mathcal{B}^\perp)$.*

Proof. The inclusion $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B} \multimap \mathcal{A})$ was shown in Ex. 5.9.(i). For the inclusion $\mathcal{L}(\mathcal{B} \multimap \mathcal{A}) \subseteq \mathcal{L}(\mathcal{B}^\perp)$, by Ex. 5.9.(ii) we can derive the sequent

$$\mathcal{B} \otimes (\mathcal{B} \multimap \mathcal{A}) \vdash \mathcal{A} \otimes \mathcal{B}$$

[16] More precisely, this is direction ($\Leftarrow$) in the proof of Prop. 5.12.

and it follows from adequacy (in the form of Prop. 5.6) that there is a winning P-strategy

$$\sigma \quad : \quad \mathcal{B} \otimes (\mathcal{B} \multimap \mathcal{A}) \quad \longrightarrow \quad \mathcal{A} \otimes \mathcal{B}$$

But now, since $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B}) = \emptyset$, it follows from Prop. 7.7 that there is a winning P-strategy

$$\tau \quad : \quad \mathcal{A} \otimes \mathcal{B} \quad \longrightarrow \quad \perp\!\!\!\perp$$

so that

$$\tau \circ \sigma \quad : \quad \mathcal{B} \otimes (\mathcal{B} \multimap \mathcal{A}) \quad \longrightarrow \quad \perp\!\!\!\perp$$

It then follows from Prop. 5.4 (monoidal closure) that there is a winning P-strategy in

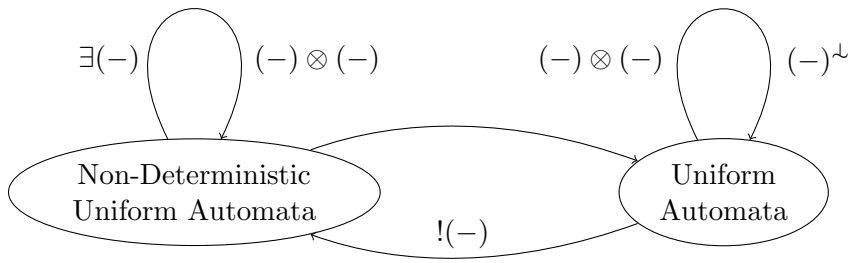
$$(\mathcal{B} \multimap \mathcal{A}) \quad \longrightarrow \quad \mathcal{B}^{\perp\!\!\!\perp}$$

and we conclude by Prop. 4.11. □

7.2. Simulation and the Exponential Modality of IMELL

Recall that similarly as in the usual setting, uniform automata have linear complements (§5.4), and that non-deterministic automata have correct existential quantifications (§7.5). On the other hand, we mentioned in §1.2 that in the usual setting, the *Simulation Theorem* [MS87, EJ91, MS95] says that each alternating automaton $\mathcal{A}$ can be simulated by a non-deterministic automaton $!\mathcal{A}$ (of exponential size) with $\mathcal{L}(!\mathcal{A}) = \mathcal{L}(\mathcal{A})$.

We show here that in our setting, an easy adaptation of the construction used in [Wal02] gives a similar simulation operation $!(-)$, taking a uniform automaton $\mathcal{A} : \Sigma$ to a non-deterministic automaton $!\mathcal{A} : \Sigma$ with $\mathcal{L}(!\mathcal{A}) = \mathcal{L}(\mathcal{A})$, thus completing the picture (2) of §1.2 for our notion of uniform automata:



(38)

Moreover, we show that the operation $!(-)$ satisfies the *deduction* rules of the exponential modality ! of IMELL:

$$\frac{M \;;\; !\overline{\mathcal{A}} \vdash \mathcal{A}}{M \;;\; !\overline{\mathcal{A}} \vdash !\mathcal{A}} \qquad \frac{M \;;\; \overline{\mathcal{A}}, \mathcal{B} \vdash \mathcal{A}}{M \;;\; \overline{\mathcal{A}}, !\mathcal{B} \vdash \mathcal{A}} \qquad \frac{M \;;\; \overline{\mathcal{A}}, \vdash \mathcal{A}}{M \;;\; \overline{\mathcal{A}}, !\mathcal{B} \vdash \mathcal{A}} \qquad \frac{M \;;\; \overline{\mathcal{A}}, !\mathcal{B}, !\mathcal{B} \vdash \mathcal{A}}{M \;;\; \overline{\mathcal{A}}, !\mathcal{B} \vdash \mathcal{A}} \tag{39}$$

It follows that the exponential ! makes it possible to define, using Girard's decomposition, an intuitionistic implication $(-) \to (-)$ as $\mathcal{A} \to \mathcal{B} := !\mathcal{A} \multimap \mathcal{B}$.

The rules (39) are an obvious adaptation to our context of the rules displayed in (24) and (25) of §2.5. The last two rules (weakening and contraction) actually follow from the rules ($\textsc{Weak}_{\textsc{ND}}$) and ($\textsc{Contr}_{\textsc{ND}}$) displayed in (37). The second rule ($\textsc{Dereliction}$) will easily follow from the construction of $!\mathcal{A}$. The most difficult rule is the first one ($\textsc{Promotion}$), which is moreover not compatible with cut-elimination (in the sense of Rem. 5.7).

The difficulty with the (Promotion) rule can be explained as follows. We have seen in §7.1 above that the symmetric monoidal structure of $\mathsf{DialAut}_\Sigma$ is Cartesian on non-deterministic automata, in other words that non-deterministic automata have a canonical comonoid structure (36). It follows that similarly as with usual IMELL-exponentials (see §2.5 but also [Mel09]), the simulation operation $!(-)$ adds to an arbitrary automaton $\mathcal{A}$ the structure allowing $!\mathcal{A}$ to be equipped with canonical maps:

$$!\mathcal{A} \quad \multimap \quad !\mathcal{A} \otimes !\mathcal{A} \qquad \text{and} \qquad !\mathcal{A} \quad \multimap \quad \mathbf{I}$$

On the other hand, recall from §5.3 that for a uniform automaton $\mathcal{A}$ with set of O-moves X, realizers of

$$\mathcal{A} \quad \multimap \quad \mathcal{A} \otimes \mathcal{A}$$

may not exist because O can play two different $(x, x') \in X \times X$ in the right component $\mathcal{A} \otimes \mathcal{A}$, that P may not be able to merge into a single $x'' \in X$ in the left component $\mathcal{A}$.

Usual solutions to this merging problem for IMELL-exponentials (see e.g. [Mel09, AC98, Mel04]) amount to equip objects of the form $!A$ with some duplication and memory abilities, essentially allowing $!A$ to run several copies of A in parallel. However (and this is via (3) §1.2, the crux of Rabin's Theorem [Rab69]), such recipes cannot (at least in an obvious way) be applied to automata on infinite trees, because $!\mathcal{A}$ must be a finite-state automaton, while plays in acceptance games (which are infinite) would require an infinite memory.

Phrased in modern terms, the solution is given by the existence of *positional* (*i.e.* memoryless) winning strategies in ω-regular games equipped with *parity* acceptance conditions (see e.g. [Tho97, GTW02]). In our case, we rely for the (Promotion) rule on the stronger fact that in an ω-regular game whose winning condition is given by a disjunction of parity conditions (also called a *Rabin* condition), winning P-strategies can always be assumed to be positional [Kla94, KK95, Jut97, Zie98]. Unfortunately, positionality is not preserved by composition, and the interpretation of the (Promotion) rule is not preserved by cut-elimination (in the sense of Rem. 5.7).

Remark 7.9. *In (38), we have only displayed existential quantifications $\exists$ for non-deterministic automata, because as in the usual setting, they are correct (in the sense of Cor. 7.5) only on non-deterministic automata. Similarly, we have not displayed universal quantifications because they are only complete on* universal *automata (see Def. 3.1).*

Note that on the other hand, the categorical properties of quantifications (Prop. 6.2) and thus the deduction rules of Fig. 20, hold on general uniform automata.

7.2.1. Parity Automata. Similarly as in the usual setting, we say that $\mathcal{A}$ is a *parity* automaton if $\Omega_\mathcal{A}$ is generated from a map $c_\mathcal{A} : Q_\mathcal{A} \to \mathbb{N}$ as the set of sequences $(q_k)_k$ such that the maximal number occurring infinitely often in $(c_\mathcal{A}(q_k))_k$ is even.

Proposition 7.10. *For every automaton $\mathcal{A} : \Sigma$, there is a parity automaton $\mathcal{A}^\dagger : \Sigma$ such that $\mathcal{A}^\dagger \simeq \mathcal{A}$ in $\mathsf{DialAut}_\Sigma$.*

Note that $\mathcal{A} \simeq \mathcal{A}^\dagger$ implies $\mathcal{L}(\mathcal{A}) = \mathcal{L}(\mathcal{A}^\dagger)$ by Prop. 4.11.

Proof of Prop. 7.10. Recall (from e.g. [Tho97, GTW02, PP04]) that every ω-regular language L can be recognized by a deterministic ω-word parity automaton $(Q_L, q_L^\imath, \partial_L, c_L)$. Following [Wal02], given $\mathcal{A} : \Sigma$ with set of P-moves U and set of O-moves X, let

$$\mathcal{A}^\dagger \quad := \quad (Q_\mathcal{A} \times Q_L \,,\, (q_\mathcal{A}^\imath, q_L^\imath) \,,\, U \,,\, X \,,\, \partial_{\mathcal{A}^\dagger} \,,\, \Omega_{\mathcal{A}^\dagger})$$

where L is the ω-regular language $\Omega_{\mathcal{A}}$, the acceptance condition $\Omega_{\mathcal{A}^\dagger}$ is generated from c_L via second projection, and the transition function $\partial_{\mathcal{A}^\dagger}$ is given by:

$$\partial_{\mathcal{A}^\dagger}((q_{\mathcal{A}}, q_L), \mathsf{a}, u, x, d) \quad := \quad (q'_{\mathcal{A}}\,,\, \partial_L(q_L, q'_{\mathcal{A}}))$$

with $q'_{\mathcal{A}} := \partial_{\mathcal{A}}(q_{\mathcal{A}}, \mathsf{a}, u, x, d)$. Note that $\mathcal{A}$ and $\mathcal{A}^\dagger$ have the same P and O-moves, so that identity strategies provide an isomorphism $\mathcal{A} \simeq \mathcal{A}^\dagger$. □

7.2.2. An Exponential Construction on Uniform Automata. Our exponential construction $!(-)$ is an adaptation of the one used in [Wal02]. Given a parity automaton $\mathcal{A} : \Sigma$ with set of P-moves U and set of O-moves X, we let

$$!\mathcal{A} \quad := \quad (Q_{!\mathcal{A}}\,,\, q^\iota_{!\mathcal{A}}\,,\, U^{Q_{\mathcal{A}}}\,,\, \mathbf{1}\,,\, \partial_{!\mathcal{A}}\,,\, \Omega_{!\mathcal{A}})$$

where $Q_{!\mathcal{A}} := \mathcal{P}(Q_{\mathcal{A}} \times Q_{\mathcal{A}})$, $q^\iota_{!\mathcal{A}} := \{(q^\iota_{\mathcal{A}}, q^\iota_{\mathcal{A}})\}$ and $\partial_{!\mathcal{A}}$ is defined as follows: Given $\mathsf{a} \in \Sigma$, $f \in U^{Q_{\mathcal{A}}}$, $d \in \mathfrak{D}$ and $S \in Q_{!\mathcal{A}}$ with $\pi_2(S) = \{q' \mid \exists q.\ (q, q') \in S\} = \{q_1, \ldots, q_n\}$, let

$$\partial_{!\mathcal{A}}(S, \mathsf{a}, f, \bullet, d) \quad := \quad T_1 \cup \cdots \cup T_n$$

where, for each $k \in \{1, \ldots, n\}$,

$$T_k \quad := \quad \{(q_k, q) \mid \exists x \in X.\ q = \partial_{\mathcal{A}}(q_k, \mathsf{a}, f(q_k), x, d)\}$$

Let a *trace* in an infinite sequence $(S_n)_n \in Q^\omega_{!\mathcal{A}}$ be a sequence $(q_n)_n$ such that for all n, $(q_n, q_{n+1}) \in S_{n+1}$. We let $\Omega_{!\mathcal{A}}$ be the set of sequences $(S_n)_n$ whose traces all belong to $\Omega_{\mathcal{A}}$. Note that $\Omega_{!\mathcal{A}}$ is ω-regular since $\Omega_{\mathcal{A}}$ is ω-regular (see [Wal02, §4]).

Remark 7.11. *Note that $Q_{!\mathcal{A}} = \mathcal{P}(Q \times Q)$ contains a "true" state $\emptyset \in Q_{!\mathcal{A}}$, so the map*

$$\tilde{\partial}_{!\mathcal{A}} \quad : \quad Q_{!\mathcal{A}} \times \Sigma \quad \longrightarrow \quad U^Q \quad \longrightarrow \quad (\mathfrak{D} \longrightarrow Q_{!\mathcal{A}})$$

is always total.

For a uniform automaton $\mathcal{A}$ whose acceptance condition is not a parity condition, let $!\mathcal{A} := !(\mathcal{A}^\dagger)$, where $\mathcal{A}^\dagger$ is obtained from Prop. 7.10.

It is easy to show the adequacy of the dereliction rule. This amounts to provide co-unit-like winning P-strategies

$$\varepsilon \quad : \quad !\mathcal{A}(M) \quad \multimap \quad \mathcal{A}(M)$$

Proposition 7.12. *Given $\mathcal{A} : \Sigma$, there is a winning P-strategy ε in $\Sigma \vdash !\mathcal{A}(M) \multimap \mathcal{A}(M)$.*

Proof. By Prop. 7.10, we can assume $\mathcal{A}$ to be a parity automaton. Using the injectivity of HS (Lem. A.11), we define $\mathrm{HS}(\varepsilon)$ by induction on plays as follows, with the following invariant: for each $(s, t) \in \mathrm{HS}(\varepsilon)$, with s, t of even length, writing q for the state of t and S for the state of s, we have $q \in \pi_2(S)$.

The base case is trivial. Let $(s, t) \in \mathrm{HS}(\varepsilon)$ with s and t even-length, and with t in state q and s in state S. Given an O-move (a, h), we let $(s.(\mathsf{a}, h)\,,\, t.h(q)) \in \mathrm{HS}(\varepsilon)$, and for all (x, d) we further let $(s.(\mathsf{a}, h).(\bullet, d)\,,\, t.h(q).(x, d)) \in \mathrm{HS}(\varepsilon)$. Then the invariant is insured by def. of $!\mathcal{A}$.

The strategy ε is winning since the sequence of states produced in $\mathcal{A}$ is a trace in the sequence of states produced in $!\mathcal{A}$. □

	$((p\,,\,\bar{\mathsf{a}}\,,\,q_{\mathcal{A}})$	$,\quad (p\,,\,\bar{\mathsf{a}}\,,\,q_{\mathcal{B}}))$	$A_{\mathsf{P}} \times B_{\mathsf{P}}$
O	$\downarrow$		
	$((p\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q_{\mathcal{A}}\,,\,u)$	$,\quad (p\,,\,\bar{\mathsf{a}}\,,\,q_{\mathcal{B}}))$	$A_{\mathsf{O}} \times B_{\mathsf{P}}$
P		$\downarrow$	
	$((p\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q_{\mathcal{A}}\,,\,u)$	$,\quad (p\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q_{\mathcal{B}}\,,\,v))$	$A_{\mathsf{O}} \times B_{\mathsf{O}}$
O		$\downarrow$	
	$((p\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q_{\mathcal{A}}\,,\,u)$	$,\quad (p.d\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q'_{\mathcal{B}}))$	$A_{\mathsf{O}} \times B_{\mathsf{P}}$
P	$\downarrow$		
	$((p.d\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q'_{\mathcal{A}})$	$,\quad (p.d\,,\,\bar{\mathsf{a}}.\mathsf{a}\,,\,q'_{\mathcal{B}}))$	$A_{\mathsf{P}} \times B_{\mathsf{P}}$

Figure 23: The edges of the graph G for $\Sigma \vdash \mathcal{A}(M) \multimap \mathcal{B}(N)$

7.2.3. Game Graphs and Positionality. We now turn to the (Promotion) rule. Its adequacy relies on well-known but non-trivial results on the existence of winning positional P-strategies for *Rabin* games, which are games whose winning conditions are disjunctions of parity conditions. The notion of *positional* strategy makes sense for games whose moves and winning condition are induced in an appropriate way by a given graph.

Consider uniform substituted acceptance games $\Sigma \vdash \mathcal{A}(M)$ and $\Sigma \vdash \mathcal{B}(N)$, where $\mathcal{A}$ (resp. $\mathcal{B}$) has set of P-moves U (resp. V) and set of O-moves X (resp. Y). The *game graph* of $\Sigma \vdash \mathcal{A}(M) \multimap \mathcal{B}(N)$ is the graph G with vertices:

$$(A_{\mathsf{P}} \times B_{\mathsf{P}}) + (A_{\mathsf{O}} \times B_{\mathsf{P}}) + (A_{\mathsf{O}} \times B_{\mathsf{O}})$$

where

$$\begin{array}{rclcrcl} A_{\mathsf{P}} & := & \mathfrak{D}^* \times \Sigma^* \times Q_{\mathcal{A}} & \qquad & A_{\mathsf{O}} & := & \mathfrak{D}^* \times \Sigma^* \times Q_{\mathcal{A}} \times U \\ B_{\mathsf{P}} & := & \mathfrak{D}^* \times \Sigma^* \times Q_{\mathcal{B}} & & B_{\mathsf{O}} & := & \mathfrak{D}^* \times \Sigma^* \times Q_{\mathcal{B}} \times V \end{array}$$

and with edges depicted in Fig. 23, where $q'_{\mathcal{A}} := \partial_{\mathcal{A}}(q_{\mathcal{A}}, M(\bar{\mathsf{a}}.\mathsf{a}, p), u, x, d)$ (for some $x \in X$) and $q'_{\mathcal{B}} := \partial_{\mathcal{B}}(q_{\mathcal{B}}, N(\bar{\mathsf{a}}.\mathsf{a}, p), v, y, d)$ (for some $y \in Y$). Write pos for the graph morphism from the set of plays of $\Sigma \vdash \mathcal{A}(M) \multimap \mathcal{B}(N)$ (seen as a tree) to G. We say that a strategy σ is *positional* if it agrees on plays with the same position, *i.e.* if $s.m \in \sigma$, $t.m' \in \sigma$ with $\mathsf{pos}(s) = \mathsf{pos}(t)$ implies $m = m'$.

Consider now parity automata $\mathcal{A}_1, \ldots, \mathcal{A}_n$ and $\mathcal{B}$. The winning condition of a game of the form $\mathcal{A}_1(M_1) \otimes \ldots \otimes \mathcal{A}_n(M_n) \multimap \mathcal{B}(N)$ is a disjunction of parity conditions, also called a *Rabin* condition, which is induced by colorings depending only on the vertices of its game graph G. It has been shown in [Kla94, KK95, Jut97, Zie98] that if P has a winning strategy σ in such a game, then it has a winning *positional* strategy (w.r.t. G), which according to [Zie98] is recursive in σ. The existence of winning positional P-strategies allows us to show the adequacy of the (Promotion) rule. The proof is deferred to App. B.

Proposition 7.13. *Given $\mathcal{N}, \mathcal{A} : \Sigma$ with $\mathcal{N}$ non-deterministic, if there is a winning* P*-strategy in $\Sigma \vdash \mathcal{N}(L) \multimap \mathcal{A}(M)$, then there is a winning* P*-strategy in $\Sigma \vdash \mathcal{N}(L) \multimap !\mathcal{A}(M)$.*

Remark 7.14. *Consider the case of $!\mathcal{A}$ for $\mathcal{A}$ non-deterministic. Given $S \in Q_{!\mathcal{A}}$ with $\pi_2(S) = \{q_1, \ldots, q_n\}$, we have*

$$\partial_{!\mathcal{A}}(S, \mathsf{a}, f, \bullet, d) \quad = \quad \{(q_k, \partial_{\mathcal{A}}(q_k, \mathsf{a}, f(q_k), \bullet, d)) \mid k = 1, \ldots, n\}$$

One can easily see that $\mathcal{A}$ is a retract of $!\mathcal{A}$. Moreover, since the initial state of $!\mathcal{A}$ is the singleton $q^{\iota}_{!\mathcal{A}} = \{(q^{\iota}_{\mathcal{A}}, q^{\iota}_{\mathcal{A}})\}$, plays in games of the form $!\mathcal{A}(M)$ only reach singleton states $\{(q_{\mathcal{A}}, q'_{\mathcal{A}})\} \in Q_{!\mathcal{A}}$. In particular, each play on $!\mathcal{A}(M)$ determines a unique play on $\mathcal{A}(M)$, and it follows that $!(-)$ extends to a functor on non-deterministic *automata.*

7.2.4. Applications. This paragraph gathers consequences of Props. 7.12 and 7.13, thus mirroring §7.1.1-7.1.3 and completing the picture announced in §1.3, §1.5 and §2.5. Furthermore, App. C details how two constructions from [CL08] and [SA05] can be reformulated in our setting.

First, Prop. 7.12 implies that $\mathcal{L}(!\mathcal{A}) \subseteq \mathcal{L}(\mathcal{A})$, while Prop. 7.13 gives the converse inclusion $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(!\mathcal{A})$. We thus have, as expected:

Corollary 7.15. $\mathcal{L}(\mathcal{A}) = \mathcal{L}(!\mathcal{A})$.

Corollary 7.15 gives the extension of Cor. 7.6 to general uniform automata.

Corollary 7.16. *Given a uniform automaton $\mathcal{A}$, one can decide whether $\mathcal{L}(\mathcal{A})$ is empty. Moreover, if $\mathcal{L}(\mathcal{A}) \neq \emptyset$ then one can effectively build from $\mathcal{A}$ a regular tree $T \in \mathcal{L}(\mathcal{A})$ together with a finite state winning* P*-strategy on* $\mathbf{1} \vdash \mathbf{I} \multimap \mathcal{A}(\dot{T})$.

We also obtain the lifting property of §1.5.(3), extending Prop. 7.7. Let $?\mathcal{A} := (!\mathcal{A}^{\perp})^{\perp}$.

Proposition 7.17 (Weak Completeness)**.** *Given automata $\mathcal{A}, \mathcal{B} : \Sigma$, if $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})$ then there is an effective winning* P*-strategy in* $\Sigma \vdash !\mathcal{A} \multimap ?\mathcal{B}$.

Proof. By Prop. 5.12 and Cor. 7.15, if $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{B})$ then $\mathcal{L}(!\mathcal{A}) \cap \mathcal{L}(!(\mathcal{B}^{\perp})) = \emptyset$, and we conclude by Prop. 7.7. □

Proposition 7.17 is a completeness result on realizability w.r.t. language inclusion. It is only a weak converse to the soundness of realizability w.r.t. language inclusion (Prop. 4.11, §4.5), because it imposes constraints on the *shape* of automata for the implication to be realizable (while it imposes no constraint on the *languages* involved as $\mathcal{L}(\mathcal{A}) = \mathcal{L}(!\mathcal{A})$ and $\mathcal{L}(\mathcal{B}) = \mathcal{L}(?\mathcal{B})$).

On the other hand, Props. 7.12 and 7.13 give adequacy for the rules displayed in (39).

Proposition 7.18 (Adequacy (Thm. 1.3 (8)))**.** *If the sequent $M \; ; \; \mathcal{A}_1, \ldots, \mathcal{A}_n \vdash \mathcal{B}$ is derivable using the rules of Fig. 18, Fig. 20 Fig. 24 and of Ex. 5.9, then there is a winning* P*-strategy in the game*

$$\mathcal{A}_1(M) \otimes_{\mathsf{DA}} \cdots \otimes_{\mathsf{DA}} \mathcal{A}_n(M) \quad \multimap \quad \mathcal{B}(M)$$

As an example of use of the exponential rules, we mention a negative translation of the law of Peirce $((A \to B) \to A) \to A$. The law of Peirce gives full classical logic when added to intuitionistic logic. Recall that $\mathcal{A} \to \mathcal{B} := !\mathcal{A} \multimap \mathcal{B}$.

Example 7.19. *The law of Peirce $((?\mathcal{A} \to ?\mathcal{B}) \to ?\mathcal{A}) \to ?\mathcal{A}$ can be derived thanks to the exponential rules.*

Proof. We can derive

$$!\mathcal{A}^{\perp}, ?\mathcal{A} \vdash \perp$$

so that (since $?\mathcal{B} = (!\mathcal{B}^{\perp})^{\perp}$)

$$!\mathcal{A}^{\perp}, ?\mathcal{A} \vdash ?\mathcal{B}$$

from which it follows that

$$!((?\mathcal{A} \to ?\mathcal{B}) \to ?\mathcal{A}), !\mathcal{A}^{\perp} \vdash ?\mathcal{A}$$

and thus

$$!((?\mathcal{A} \to ?\mathcal{B}) \to ?\mathcal{A}), !\mathcal{A}^{\perp} \vdash \perp$$

and we are done since $?\mathcal{A} = (!\mathcal{A}^{\perp})^{\perp}$. □

$$(\textsc{Dereliction})\ \frac{M\ ;\ \overline{\mathcal{A}}, \mathcal{A}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}}, !\mathcal{A}, \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{N}} \vdash \mathcal{A}}{M\ ;\ \overline{\mathcal{N}} \vdash !\mathcal{A}}\ (\textsc{Promotion})$$

$$(\textsc{Weak}_{\textsc{nd}})\ \frac{M\ ;\ \overline{\mathcal{A}}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}}, \mathcal{N}, \overline{\mathcal{B}} \vdash \mathcal{C}} \qquad \frac{M\ ;\ \overline{\mathcal{A}}, \mathcal{N}, \mathcal{N}, \overline{\mathcal{B}} \vdash \mathcal{C}}{M\ ;\ \overline{\mathcal{A}}, \mathcal{N}, \overline{\mathcal{B}} \vdash \mathcal{C}}\ (\textsc{Contr}_{\textsc{nd}})$$

Figure 24: Exponential rules (where $\mathcal{N}$ and $\overline{\mathcal{N}}$ are non-deterministic)

Finally, returning to MSO and IMELL (in the sense of §1.1 and §1.3), we obtain Prop. 1.1 As in §1.3, we assume given an automaton $\mathcal{A}(\alpha)$ for each atomic formula $\alpha \in \mathrm{At}$.

Proposition 7.20 (Prop. 1.1)**.** *Consider a closed MSO-formulae φ as in §1.1, and let $(-)^\dagger$ be either $(-)^{\mathrm{nd}}$ or $(-)^{\mathrm{alt}}$. Then φ is true (in the standard model) if and only if $\mathcal{A}(\varphi^\dagger)$ accepts the unique $\mathbf{1}$-labeled tree.*

Proof. Given an MSO-formula φ with free variables among $X_1, \ldots, X_p$, for $T : \mathfrak{D}^* \to \mathbf{2}^p$, we have

$$T \models \varphi \qquad \Longleftrightarrow \qquad T \in \mathcal{L}(\mathcal{A}(\varphi^\dagger)) \qquad\qquad (\text{modulo } (\mathfrak{D}^* \to \mathbf{2}^p) \simeq (\mathfrak{D}^* \to \mathbf{2})^p)$$

The proof is by induction on φ, and follows from Prop. 5.10, Prop. 5.12, Cor. 7.5 and Cor. 7.15. □

8. Conclusion

We have presented preliminary results toward a Curry-Howard approach to automata on infinite trees. Our contributions concern mainly two related directions.

First, we have shown that the operations on tree automata used in the translations of MSO-formulae to automata underlying Rabin's Theorem [Rab69] correspond to the connectives of IMELL [Gir87]. Namely, we equipped a variant of usual alternating tree automata (that we called *uniform* tree automata, §3) with a fibred monoidal closed structure (§4 and §5), which in particular handles a conjunction and, via game determinacy, a linear complementation of alternating automata, as well as deduction rules for existential and universal quantifications (§6). Moreover, we have shown in §7 that this monoidal structure is Cartesian on non-deterministic automata, and in particular that (an adaptation of) a usual powerset construction for the Simulation Theorem [MS87, EJ91, MS95] satisfies the *deduction rules* of an $!(-)$ IMELL-exponential modality.

Second, our approach is based on a realizability semantics for a linear constructive deduction system on tree automata, in which, thanks to the monoidal-closed structure, realizers are winning strategies in a generalization of acceptance games. Our realizability semantics satisfies an expected property of witness extraction from proofs of existential statements. Moreover, this realizability semantics is compositional and makes it possible to combine realizers produced as interpretations of proofs with strategies witnessing (non-)emptiness of tree automata, possibly obtained using external algorithms.

We believe that this can provide a basis for semi-automatic approaches to MSO on infinite trees[17], in which, similarly as with interactive proof systems, decision algorithms can be com-

[17] Even if there are numerous implementations of decision algorithms on *tree automata*, we are aware of no working implementation of decision procedures for the full language of MSO on infinite trees.

bined with human-produced proofs or proof-search techniques. The author and P. Pradic have recently obtained preliminary results in this direction for MSO on ω-words [PR17, PR18].

Furthermore, as shown in Ex. 6.4 (see also Rem. 3.11), our interpretation shares a formal similarity with Gödel's *Dialectica* interpretation (see e.g. [AF98, Koh08]). Actually, the category **DZ** can be constructed (via a distributive law) from a category of *simple self dualization* [HS99, HS03] (over the topos of trees, see e.g. [BMSS12]), which can be seen as a skeleton of Dialectica-like categories [dP91], and the category DialZ has a shape similar to Dialectica fibrations (see [Hyl02, Hof11] but also [Jac01, Ex. 1.10.11]). This connection has been made precise in [PR19] in the case of ω-words and provides realizers for linear variants of Markov and choice rules.

Moreover, we show in App. C that our setting easily handles known constructions from [CL08] and [SA05] for language reduction and separation.

8.1. Further Works. We plan to continue the line of research initiated here and in [Rib15] along different directions. A central point w.r.t. most of them concerns the (Promotion) rule.

The interpretation of Simulation as an $!(-)$ IMELL-exponential modality in §7.2 is interesting because it shows that an IMELL-like exponential arises precisely where there is a semantic difficulty (positionality) together with a non-trivial exponential construction on automata. However, we find the interpretation of the (Promotion) rule in §7.2 not completely satisfactory for the following reasons.

(1) We have to rely on the *external* result that winning P-strategies can always be assumed to be positional in Rabin games [Kla94, KK95, Jut97, Zie98]. There seems to be essentially two ways to apply this result: (a) one could try to *extract* the positional strategy realizing the conclusion of (Promotion) from the realizer of the premise, or (b) one could obtain the strategy for the conclusion from an algorithm solving ω-regular games (that is from the Büchi-Landweber Theorem [BL69], see also e.g. [Tho97, Thm. 6.16]).

    However, in both cases this amounts to apply a non-trivial external algorithm, and there seems to be no obvious structural relation between the realizer of the conclusion and the realizer of the premise.

(2) This interpretation of the (Promotion) rule is not compatible with cut-elimination (in the sense of Rem. 5.7), because the notion of positionality required for (Promotion) is not preserved by composition, so that $!(-)$ is not a functor.

    It is unclear to us whether this is a true drawback, because we can still compose realizers and extract witnesses for existentials (§7.1.2). The only point is that two derivations which are equal modulo cut-elimination may be interpreted by two different strategies. But still, the non-functoriality of $!(-)$ is somehow uncomfortable from a semantic perspective.

First, we plan to pursue some work on the category **DZ** of zig-zag games in order to get a better picture of its usual game semantics exponentials. According to the discussion of §7.2, such exponentials would involve some infinite memory, because plays are infinite in **DZ**. Moreover, it seems reasonable to target some relaxation of **DZ** with finite limits (typically by allowing games to be equipped with a notion of legal plays).

(1) Taking inspiration from [MTT09], We plan to investigate the existence of free exponentials in suitable extensions of **DZ**.

(2) Moreover, there seems to be a natural exponential, in which P essentially plays strategies, but which in the context of automata would lead to infinite state automata.

(3) We also plan to look at non-synchronous exponentials, such as the Curien-Lamarche exponential of *sequential data structures* (see e.g. [AC98, Chap. 14], but also [Mel05]), in particular because of its backtracking abilities. We suspect that this could allow to handle known results and constructions for reduction and separation properties, in the vein of [Arn99, AN07, FMS13]. However, we do not know yet if this can provide new results.

Second, an important direction of future work is to get a better semantic account of the notion of positionality used in the interpretation of the (Promotion) rule. In the realm of game semantics, it has been shown by Melliès [Mel06] that the notion of *Innocence* (originally introduced by [HO00] via a notion of pointers on moves), which characterizes a form of functional (state-free) behavior, corresponds to some notion of positionality. Innocence is actually a strong form of positionality, which is preserved by composition. It is possible to equip DialAut-games with an obvious notion of pointers, representing applications of the transition function of automata as unfoldings of fixpoints. This leads via innocence to a notion of positionality which seems to be equipped with a monoidal-closed structure (w.r.t. to the synchronous direct product of automata), but which seems too restrictive to handle strategies obtained (via Büchi-Landweber Theorem) from emptiness checking in the sense of Cor. 6.5, §7.1.3, Cor. 7.16, and Prop. 7.17. On the other hand, the notion of positionality used in §7.2.3 may be preserved by composition for *non-deterministic* innocent strategies, in the vein of [HP12, TO15]. We do not know yet how such notions of non-deterministic strategies behave w.r.t. the construction of positional winning P-strategies for Rabin games as in e.g. [Zie98]. Also, the present setting has still to be compared with Melliès's *Higher-Order Automata* [Mel17].

Our main target is the construction of realizability models for MSO. In the case of ω-words (that is taking $\mathfrak{D} = \mathbf{1}$ in this paper), and in the context of Church's synthesis, the aforementioned results of [PR17, PR18] suggest that, together with the results of this paper, it is possible and pertinent to devise refinements of MSO based on *Intuitionistic Linear Logic* (ILL). We also already mentioned above the connection with Gödel's *Dialectica* interpretation, which suggests that it may be possible to realize linear variants of Markov and choice rules. Furthermore, this paper indicates that working in a linear deduction system for MSO allows for a fibred monoidal closed structure, with in particular deduction rules for existential and universal quantifications. We think that this can provide a good basis to handle some axioms of MSO, and moreover that ILL can provide classes of formulae with improved translations to automata w.r.t. the known non-elementary lower bound (see e.g. [GTW02, Chap. 13]).

Moreover, in devising realizability models for MSO, and in particular following the approach of this paper which decomposes the translation of formulae to automata using linear logic, a crucial role is played by the logical interpretation of the (Promotion) rule. Following [Möl02], it seems that (Promotion) may be seen as a form of reflection scheme. Similarly as in the complementation construction of [Tho97, Thm. 6.9], such reflection scheme would simply say that, because they can be assumed to be positional, realizers can be seen as labeled $\mathfrak{D}$-ary trees. This would simply amount to the fact that predicates of the form $\exists\sigma(\sigma : \mathcal{A} \multimap \mathcal{B})$ are definable in MSO.

References

[Abr97] S. Abramsky. Semantics of Interaction. In A. M. Pitts and P. Dybjer, editors, *Semantics and Logics of Computation*, volume 14 of *Publications of the Newton Institute*, page 1. Cambridge University Press, Cambridge, 1997. 2, 20, 24, 73, 76, 77, 81, 83

[AC98] R. M. Amadio and P.-L. Curien. *Domains and Lambda-Calculi*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, Cambridge, 1998. 2, 12, 19, 20, 24, 25, 61, 67, 73, 77

[AF98] J. Avigad and S. Feferman. Gödel's functional ("Dialectica") interpretation. In S. Buss, editor, *Handbook of Proof Theory*, volume 137 of *Studies in Logic and the Foundations of Mathematics*, pages 337–405. Elsevier, Amsterdam, 1998. 25, 52, 66

[AN07] A. Arnold and D. Niwinski. Continuous Separation of Game Languages. *Fundam. Inform.*, 81(1-3):19–28, 2007. 67

[Arn99] A. Arnold. The μ-calculus alternation-depth hierarchy is strict on binary trees. *ITA*, 33(4/5):329–340, 1999. 67

[BdRV02] P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, Cambridge, 2002. 1

[BE93] A. Bucciarelli and T. Ehrhard. A Theory of Sequentiality. *Theoretical Computer Science*, 113(2):273–291, 1993. 24, 25

[BGG97] E. Börger, E. Grädel, and Y. Gurevich. *The Classical Decision Problem*. Perspectives in Mathematical Logic. Springer-Verlag Berlin Heidelberg, 1997. 1

[BL69] J. R. Büchi and L. H. Landweber. Solving Sequential Conditions by Finite-State Strategies. *Transation of the American Mathematical Society*, 138:367–378, 1969. 9, 53, 58, 59, 66

[Blu13] A. Blumensath. An algebraic proof of Rabin's Tree Theorem. *Theor. Comput. Sci.*, 478:1–21, 2013. 2

[BMSS12] L. Birkedal, R. E. Møgelberg, J. Schwinghammer, and K. Støvring. First steps in synthetic guarded domain theory: step-indexing in the topos of trees. *Logical Methods in Computer Science*, 8(4):1–45, 2012. 25, 66

[CL08] T. Colcombet and C. Löding. The Non-deterministic Mostowski Hierarchy and Distance-Parity Automata. In *Automata, Languages and Programming (ICALP 2008)*, volume 5126 of *Lecture Notes in Computer Science*, pages 398–409. Springer, Berlin, Heidelberg, 2008. 10, 15, 55, 64, 66, 86, 87

[dP91] V. de Paiva. The Dialectica categories. Technical Report 213, University of Cambridge Computer Laboratory, January 1991. 25, 66

[EJ91] E. A. Emerson and C. S. Jutla. Tree Automata, Mu-Calculus and Determinacy (Extended Abstract). In *32nd Annual Symposium of Foundations of Computer Science (FOCS 1991)*, pages 368–377. IEEE Computer Society, Washington, DC, USA, 1991. 4, 5, 20, 55, 60, 65

[FMS13] A. Facchini, F. Murlak, and M. Skrzypczak. Rabin-Mostowski Index Problem: A Step beyond Deterministic Automata. In *28th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2013)*, pages 499–508. IEEE Computer Society, Washington, DC, USA, 2013. 67

[GH82] Y. Gurevich and L. Harrington. Trees, Automata, and Games. In *14th Annual ACM Symposium on Theory of Computing (STOC 1982)*, pages 60–65. ACM, New York, NY, USA, 1982. 3, 4

[Gir87] J.-Y. Girard. Linear Logic. *Theoretical Computer Science*, 50:1–102, 1987. 2, 19, 55, 65

[GLT89] J.-Y. Girard, Y. Lafont, and P. Taylor. *Proofs and Types*. Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, Cambridge, 1989. 2

[GTW02] E. Grädel, W. Thomas, and T. Wilke, editors. *Automata, Logics, and Infinite Games: A Guide to Current Research*, volume 2500 of *Lecture Notes in Computer Science*. Springer-Verlag Berlin Heidelberg, 2002. 2, 61, 67, 74, 75

[HHM07] R. Harmer, M Hyland, and P.-A. Melliès. Categorical combinatorics for innocent strategies. In *22nd Annual IEEE Symposium on Logic in Computer Science (LICS 2007)*, pages 379–388. IEEE Computer Society, Washington, DC, USA, 2007. 30

[HO00] J. M. E. Hyland and C.-H. Ong. On Full Abstraction for PCF: I, II, and III. *Information and Computation*, 163(2):285–408, 2000. 67, 76

[Hof11] P. J. W. Hofstra. The dialectica monad and its cousins. In M. Makkai and B.T. Hart, editors, *Models, Logics, and Higher-dimensional Categories: A Tribute to the Work of Mihály Makkai*, CRM proceedings & lecture notes. American Mathematical Society, Providence, Rhode Island, USA, 2011. 25, 31, 66

[HP12] T. Hirschowitz and D. Pous. Innocent Strategies as Presheaves and Interactive Equivalences for CCS. *Sci. Ann. Comp. Sci.*, 22(1):147–199, 2012. 67

[HS99] J. M. E. Hyland and A. Schalk. Abstract Games for Linear Logic. *Electr. Notes Theor. Comput. Sci.*, 29:127–150, 1999. 25, 27, 29, 66, 81, 83

[HS03] J. M. E. Hyland and A. Schalk. Glueing and orthogonality for models of linear logic. *Theoretical Computer Science*, 294(1/2):183–231, 2003. 25, 27, 29, 66

[Hyl97] J. M. E. Hyland. Game Semantics. In A. M. Pitts and P. Dybjer, editors, *Semantics and Logics of Computation*, volume 14 of *Publications of the Newton Institute*, page 131. Cambridge University Press, Cambridge, 1997. 2, 20, 24, 73, 76, 77, 81, 83

[Hyl02] J. M. E. Hyland. Proof theory in the abstract. *Annals of Pure Applied Logic*, 114(1-3):43–78, 2002. 25, 31, 66

[Jac01] B. Jacobs. *Categorical Logic and Type Theory*. Studies in logic and the foundations of mathematics. Elsevier, Amsterdam, 2001. 2, 12, 25, 27, 31, 44, 50, 56, 66

[Joh86] P.T. Johnstone. *Stone Spaces*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, Cambridge, 1986. 4

[Jut97] C. S. Jutla. Determinization and Memoryless Winning Strategies. *Information and Computation*, 133(2):117–134, 1997. 61, 63, 66, 84

[KK95] N. Klarlund and D. Kozen. Rabin Measures. *Chicago Journal of Theoretical Computer Science*, 1995(3), 1995. 61, 63, 66, 84

[Kla94] N. Klarlund. Progress measures, immediate determinacy, and a subset construction for tree automata. *Annals of Pure and Applied Logic*, 69(2-3):243–268, 1994. 61, 63, 66, 84

[Koh08] U. Kohlenbach. *Applied Proof Theory: Proof Interpretations and their Use in Mathematics*. Springer Monographs in Mathematics. Springer-Verlag Berlin Heidelberg, 2008. 8, 25, 52, 66

[KV05] O. Kupferman and M. Y. Vardi. Safraless decision procedures. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2005)*, pages 531–542. IEEE Computer Society, Washington, DC, USA, 2005. 2, 9

[LS86] J. Lambek and P. J. Scott. *Introduction to Higher Order Categorical Logic*. Cambridge Studies in Advanced Mathematics. Cambridge University Press, Cambridge, 1986. 2, 12

[McN66] R. McNaughton. Testing and generating infinite sequences by a finite automaton. *Information and Control*, 9(5):521 – 530, 1966. 2, 9

[Mel04] P.-A. Melliès. Comparing hierarchies of types in models of linear logic. *Information and Comput.*, 189(2):202–234, 2004. 61

[Mel05] P.-A. Melliès. Sequential algorithms and strongly stable functions. *Theoretical Computer Science*, 343(1-2):237–281, 2005. 20, 67, 73, 76

[Mel06] P.-A. Melliès. Asynchronous games 2: The true concurrency of innocence. *Theoretical Computer Science*, 358(2-3):200–228, 2006. 67, 76

[Mel09] P.-A. Melliès. Categorical semantics of linear logic. In *Interactive models of computation and program behaviour*, volume 27 of *Panoramas et Synthèses*. SMF, Paris, 2009. 2, 12, 19, 20, 29, 33, 36, 40, 46, 55, 56, 61

[Mel12] P.-A. Melliès. Game Semantics in String Diagrams. In *27th Annual IEEE Symposium on Logic in Computer Science (LICS 2012)*, pages 481–490. IEEE Computer Society, Washington, DC, USA, 2012. 76

[Mel13] P.-A. Melliès. On dialogue games and coherent strategies. In *Computer Science Logic (CSL 2013)*, volume 23 of *LIPIcs*, pages 540–562. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2013. 49

[Mel17] P.-A. Melliès. Higher-order parity automata. In *32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2017)*, pages 1–12. IEEE Computer Society, Washington, DC, USA, 2017. 67

[ML98] S. Mac Lane. *Categories for the Working Mathematician*. Springer-Verlag New York, 2nd edition, 1998. 40, 50

[MM15] K. Maillard and P.-A. Melliès. A Fibrational Account of Local States. In *30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2015)*, pages 402–413. IEEE Computer Society, Washington, DC, USA, 2015. 31

[Möl02] M. Möllerfeld. *Generalized Inductive Definitions - The μ-calculus and Π^1_2-comprehension*. PhD thesis, Westfälischen Wilhelms-Universität Münster, 2002. Available at https://miami.uni-muenster.de/Record/9dfa74b6-186b-4e95-a51f-9965d7e1e508. 67

[MS87] D. E. Muller and P. E. Schupp. Alternating Automata on Infinite Trees. *Theoretical Computer Science*, 54:267–276, 1987. 4, 5, 20, 48, 55, 60, 65

[MS95] D. E. Muller and P. E Schupp. Simulating Alternating Tree Automata by Nondeterministic Automata: New Results and New Proofs of the Theorems of Rabin, McNaughton and Safra. *Theoretical Computer Science*, 141(1&2):69–107, 1995. 4, 5, 20, 55, 60, 65

[MTT09] P.-A. Melliès, N. Tabareau, and C. Tasson. An Explicit Formula for the Free Exponential Modality of Linear Logic. In *Automata, Languages and Programming (ICALP 2009)*, volume 5556 of *Lecture Notes in Computer Science*, pages 247–260. Springer, Berlin, Heidelberg, 2009. 66

[Ong06] C.-H. L. Ong. On Model-Checking Trees Generated by Higher-Order Recursion Schemes. In *21st Annual IEEE Symposium on Logic in Computer Science (LICS 2006)*, pages 81–90. IEEE Computer Society, Washington, DC, USA, 2006. 17

[PP04] D. Perrin and J.-É. Pin. *Infinite Words: Automata, Semigroups, Logic and Games*. Pure and Applied Mathematics. Elsevier, Amsterdam, 2004. 2, 49, 61, 74, 75

[PR17] P. Pradic and C. Riba. A Curry-Howard Approach to Church's Synthesis. In *2nd International Conference on Formal Structures for Computation and Deduction (FSCD 2017)*, volume 84 of *LIPIcs*, pages 30:1–30:16. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2017. 9, 66, 67

[PR18] P. Pradic and C. Riba. LMSO: A Curry-Howard Approach to Church's Synthesis via Linear Logic. In *33rd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2018)*, pages 849–858. ACM, New York, NY, USA, 2018. 9, 17, 66, 67

[PR19] P. Pradic and C. Riba. A Dialectica-Like Interpretation of a Linear MSO on Infinite Words. In *Foundations of Software Science and Computation Structures (FoSSaCS 2019)*, Lecture Notes in Computer Science, pages 470–487. Springer, Cham, 2019. 25, 66

[Rab69] M. O. Rabin. Decidability of Second-Order Theories and Automata on Infinite Trees. *Transactions of the American Mathematical Society*, 141:1–35, 1969. 1, 2, 3, 55, 61, 65

[Rab72] M. O. Rabin. Automata on infinite objects and Church's Problem. *Amer. Math. Soc.*, 1972. 58

[Rib15] C. Riba. Fibrations of tree automata. In *13th International Conference on Typed Lambda Calculi and Applications (TLCA 2015)*, volume 38 of *LIPIcs*, pages 302–316. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2015. 2, 5, 7, 8, 10, 14, 17, 18, 19, 20, 23, 24, 25, 27, 66, 73, 76, 77, 81

[SA05] L. Santocanale and A. Arnold. Ambiguous classes in mu-calculi hierarchies. *Theoretical Computer Science*, 333(1-2):265–296, 2005. 10, 55, 64, 66, 86, 88, 91

[SU06] M. H. Sørensen and P. Urzyczyn. *Lectures on the Curry-Howard Isomorphism*, volume 149 of *Studies in Logic and the Foundations of Mathematics*. Elsevier, Amsterdam, 2006. 2, 8

[Tho97] W. Thomas. Languages, Automata, and Logic. In G. Rozenberg and A. Salomaa, editors, *Handbook of Formal Languages*, volume III, pages 389–455. Springer-Verlag Berlin Heidelberg, 1997. 2, 3, 49, 53, 58, 59, 61, 66, 67, 74, 75, 87

[TO15] T. Tsukada and C.-H. L. Ong. Nondeterminism in Game Semantics via Sheaves. In *30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2015)*, pages 220–231. IEEE Computer Society, Washington, DC, USA, 2015. 67

[VW08] M. Y. Vardi and T. Wilke. Automata: from logics to algorithms. In *Logic and Automata*, volume 2 of *Texts in Logic and Games*, pages 629–736. Amsterdam University Press, Amsterdam, 2008. 1

[Wal02] I. Walukiewicz. Monadic second-order logic on tree-like structures. *Theoretical Computer Science*, 275(1-2):311–346, 2002. 4, 49, 60, 61, 62, 84

[Zie98] Z. Zielonka. Infinite games on finitely coloured graphs with applications to automata on infinite trees. *Theoretical Computer Science*, 200(1-2):135–183, 1998. 61, 63, 66, 67, 84

A. Linear Synchronous Arrow Games

This Appendix relates our categories to [Rib15] and to standard concepts in game semantics. We also prove that $\mathbf{DZ}$ and $\mathbf{DZ}^{\mathrm{W}}$ are categories.

A.1. Simple Games. Game semantics provide models of typed λ-calculi, and can thus be used, via the Curry-Howard correspondence, to build compositional models of deduction systems. In game models, *types* (and, via Curry-Howard, formulae, or in our context automata instantiated with trees or $\mathbf{T}$-maps) are interpreted by two-player sequential games where the *Proponent* P ($\exists$loïse) and the *Opponent* O ($\forall$belard) play in turn moves, producing plays subject to specified rules. Originally, game models where proposed because they provide *fully complete* models of various λ-calculi, in the sense that finite P-strategies are definable by λ-terms. On the other hand, the notion of strategy naturally encompasses infinite objects, and is thus well suited to deal with runs of automata on infinite trees.

There are different families of game models. We use *simple games* (see e.g. [Abr97, Hyl97]), which stem from Berry & Curien's *sequential data structures* (see e.g. [AC98, Chap. 14], but also [Mel05]).

Definition A.1 (Simple Games)**.**

- *A* simple game *A has the form*

$$A \quad = \quad (A_{\mathsf{P}}\,,\, A_{\mathsf{O}}\,,\, L_A)$$

  where A_{P} is the set of P-moves*, A_{O} is the set of* O-moves *and $L_A \subseteq (A_{\mathsf{P}} + A_{\mathsf{O}})^*$ is a non-empty prefix-closed set of* legal plays*.*

  We let $s, t, \ldots$ range of over plays and $m, n, \ldots$ range over moves.

- *A* simple game with winning *is a simple game A equipped with a* set of winning plays *(or* winning condition*) $\mathcal{W}_A \subseteq (A_{\mathsf{P}} + A_{\mathsf{O}})^\omega$.*

- *The sets $\wp_A^{\mathsf{P}}$ and $\wp_A^{\mathsf{O}}$ of resp.* positive *and* negative *plays on A are*

$$\begin{aligned} \wp_A^{\mathsf{P}} \quad &:= \quad (A_{\mathsf{P}} \cdot A_{\mathsf{O}})^* + (A_{\mathsf{P}} \cdot A_{\mathsf{O}})^* \cdot A_{\mathsf{P}} \\ \wp_A^{\mathsf{O}} \quad &:= \quad (A_{\mathsf{O}} \cdot A_{\mathsf{P}})^* + (A_{\mathsf{O}} \cdot A_{\mathsf{P}})^* \cdot A_{\mathsf{O}} \end{aligned}$$

  The game A is positive *(resp.* negative*) if all its legal plays are positive (resp. negative), that is if $L_A \subseteq \wp_A^{\mathsf{P}}$ (resp. $L_A \subseteq \wp_A^{\mathsf{O}}$). So P starts in a positive game and O starts in a negative one. A game is* polarized *if it is either positive or negative.*

- *A play for player $\xi \in \{\mathsf{P}, \mathsf{O}\}$ (also called a ξ-*play*) is either the empty play or a non-empty play in which ξ played last (i.e. which ends with a ξ-move).*

Example A.2 (Full Postitive Games (Def. 3.3))**.** *In the setting of Def. A.1, the* full positive games *of Def. 3.3 are positives games A whose positive plays are all legal, that is such that*

$$L_A \quad = \quad \wp_A^{\mathsf{P}} \quad = \quad (A_{\mathsf{P}} \cdot A_{\mathsf{O}})^* + (A_{\mathsf{P}} \cdot A_{\mathsf{O}})^* \cdot A_{\mathsf{P}}$$

Hence, as in Def. 3.3, a full positive game A is completely characterized by its set of P *and* O*-moves.*

Example A.3. *In the case of a non-deterministic automaton $\mathcal{A}$ on a tree T, following the usual setting (see e.g. [Tho97, GTW02, PP04]), the acceptance game $\mathcal{G}(\mathcal{A}, T)$ is defined as the positive simple game with winning*

$$\mathcal{G}(\mathcal{A}, T) \quad := \quad (\mathfrak{D} \to Q_{\mathcal{A}},\ Q_{\mathcal{A}} \times \mathfrak{D},\ L_{\mathcal{A}(T)},\ \mathcal{W}_{\mathcal{A}(M)})$$

whose legal plays $s \in L_{\mathcal{A}(T)}$ are sequences of the form

$$\begin{array}{rrcl} & s & = & \mathsf{g}_0 \cdot (q_1, d_1) \cdot \mathsf{g}_1 \cdot \ldots \cdot (q_n, d_n) \\ \text{or} & s & = & \mathsf{g}_0 \cdot (q_1, d_1) \cdot \mathsf{g}_1 \cdot \ldots \cdot (q_n, d_n) \cdot \mathsf{g}_n \end{array}$$

where $n \geq 0$, $q_{k+1} = \mathsf{g}_k(d_{k+1})$ and $\mathsf{g}_k \in \partial_{\mathcal{A}}(q_k, T(d_1 \cdot \ldots \cdot d_k))$ with $q_0 := q^{\iota}_{\mathcal{A}}$. Note that O only chooses the tree directions $d_1, \ldots, d_n \in \mathfrak{D}$, while P chooses from each O-play

$$\mathsf{g}_0 \cdot (q_1, d_1) \cdot \mathsf{g}_1 \cdot \ldots \cdot (q_n, d_n)$$

a function $\mathsf{g}_n : \mathfrak{D} \to Q_{\mathcal{A}}$ available in $\partial_{\mathcal{A}}(q_n, T(d_1 \cdot \ldots \cdot d_n))$.

The winning plays $\chi \in \mathcal{W}_{\mathcal{A}(T)}$ are generated from the acceptance condition $\Omega_{\mathcal{A}}$ in the expected way. We let $\mathcal{W}_{\mathcal{A}(T)} \subseteq ((\mathfrak{D} \to Q_{\mathcal{A}}) \cdot (Q_{\mathcal{A}} \times \mathfrak{D}))^{\omega}$ consist of the infinite sequences

$$\chi \quad = \quad \mathsf{g}_0 \cdot (q_1, d_1) \cdot \mathsf{g}_1 \cdot \ldots \cdot (q_n, d_n) \cdot \ldots$$

such that $(q_k)_{k \in \mathbb{N}} \in \Omega_{\mathcal{A}}$ (where $q_0 := q^{\iota}_{\mathcal{A}}$).

We now come to the definition of strategies in simple games. A strategy for player P or O is what one expects. The formal definition of strategy below emphasizes P (strategies for O are defined by duality), because in categories of games, composition and identities are only defined for the strategies of the negative player P. Moreover, the manipulation of strategies as morphisms is more convenient when strategies are presented as sets of plays rather than as functions on plays.

Definition A.4 (Strategies)**.** *A* P-strategy *on A is a non-empty set of legal P-plays $\sigma \subseteq L_A$ which is*

- P***-prefix-closed:*** *if $s.t \in \sigma$ and s is a P-play then $s \in \sigma$, and*
- P***-deterministic:*** *if $s.n \in \sigma$ and $s.m \in \sigma$ then $n = m$.*

Consider now a polarized game with winning A. Given a P-strategy σ on A and an O-play $s \in L_A$, we say that s is an O-interrogation *of σ if either $s = \epsilon$ and A is positive, or if $s = t.m$ for some $t \in \sigma$. We say that σ is* total *if for every O-interrogation s of σ, we have $s.n \in \sigma$ for some n. A* winning *(P-)strategy on A is a total strategy σ s.t. for all $\chi \in (A_{\mathsf{P}} + A_{\mathsf{O}})^{\omega}$, we have $\chi \in \mathcal{W}_A$ whenever $\exists^{\infty} k \in \mathbb{N}.\ \chi(0) \cdot \ldots \cdot \chi(k) \in \sigma$.*

The notion of (total, winning) O-strategy is defined by duality. Each game A has a dual $\overline{A} = (A_{\mathsf{O}}, A_{\mathsf{P}}, L_A)$, where we moreover let $\mathcal{W}_{\overline{A}} := (A_{\mathsf{P}} + A_{\mathsf{O}})^{\omega} \setminus \mathcal{W}_A$ if A is a game with winning. Note that $\overline{A}$ is polarized iff A is polarized, and that $\overline{A}$ is positive (resp. negative) iff A is negative (resp. positive). Then, we say that a (total, winning) O-strategy on A is a (total, winning) P-strategy on $\overline{A}$.

Example A.5 (Strategies in Full Postitive Games (Def. 3.4))**.** *Consider a total full positive game $A = (U, X)$. In the setting of Def. A.4, a* P*-strategy σ in A is a non-empty set of sequences of the form*

$$s \quad = \quad u_0 \cdot x_1 \cdot u_1 \cdot \ldots \cdot x_{n-1} \cdot u_{n-1}$$

such that

$$s \cdot x_n \cdot u_n \ \in \ \sigma \quad \Longrightarrow \quad s \in \sigma$$

and

$$s \cdot x_n \cdot u_n \ , \ s \cdot x_n \cdot u'_n \ \in \ \sigma \quad \Longrightarrow \quad u_n = u'_n$$

Hence, as expected, in a play

$$u_0 \cdot x_1 \cdot u_1 \cdot \ldots \cdot x_n \cdot u_n \ \in \ \sigma$$

the moves u_i are uniquely determined by the O*-moves $x_1, \ldots, x_{i-1}$. Moreover, σ is total iff for every*

$$u_0 \cdot x_1 \cdot u_1 \cdot \ldots \cdot x_n \cdot u_n \ \in \ \sigma$$

and for every O*-move x_{n+1}, there is some u_{n+1} such that*

$$u_0 \cdot x_1 \cdot u_1 \cdot \ldots \cdot x_n \cdot u_n \cdot x_{n+1} \cdot u_{n+1} \ \in \ \sigma$$

In other words, total P*-strategies in a total full positive game $A = (U, X)$ are indeed given by functions $X^* \to U$, as in Def. 3.4.*

Example A.6. *In the case of the acceptance game $\mathcal{G}(\mathcal{A}, T)$ described above, a* P*-strategy σ is therefore a non-empty set of sequences of the form*

$$s \quad = \quad \mathsf{g}_0 \cdot (q_1, d_1) \cdot \mathsf{g}_1 \cdot \ldots \cdot (q_{n-1}, d_{n-1}) \cdot \mathsf{g}_{n-1}$$

such that

$$s \cdot (q_n, d_n) \cdot \mathsf{g}_n \ \in \ \sigma \quad \Longrightarrow \quad s \in \sigma$$

and

$$s \cdot (q_n, d_n) \cdot \mathsf{g}_n \ , \ s \cdot (q_n, d_n) \cdot \mathsf{g}'_n \ \in \ \sigma \quad \Longrightarrow \quad \mathsf{g}_n = \mathsf{g}'_n$$

Moreover, σ is total iff for every

$$\mathsf{g}_0 \cdot (q_1, d_1) \cdot \mathsf{g}_1 \cdot \ldots \cdot (q_n, d_n) \cdot \mathsf{g}_n \ \in \ \sigma$$

and for every $d_{n+1} \in \mathfrak{D}$, there is some g_{n+1} such that

$$\mathsf{g}_0 \cdot (q_1, d_1) \cdot \mathsf{g}_1 \cdot \ldots \cdot (q_n, d_n) \cdot \mathsf{g}_n \cdot (\mathsf{g}_n(d_{n+1}), d_{n+1}) \cdot \mathsf{g}_{n+1} \ \in \ \sigma$$

It follows that a total P*-strategy σ on $\mathcal{G}(\mathcal{A}, T)$ is uniquely determined by a run tree R such that $R(\epsilon) = q^\iota_{\mathcal{A}}$, and such that for every $d_1 \cdot \ldots \cdot d_n \cdot d_{n+1} \in \mathfrak{D}^{n+1}$,*

$$R(d_1 \cdot \ldots \cdot d_n \cdot d_{n+1}) \quad = \quad \mathsf{g}_n(d_{n+1})$$

where, for $0 \leq k \leq n$, the g_k are unique such that

$$\mathsf{g}_0 \cdot (\mathsf{g}_0(d_1), d_1) \cdot \ldots \cdot (\mathsf{g}_{k-1}(d_k), d_k) \cdot \mathsf{g}_k \ \in \ \sigma$$

Hence (winning) total P*-strategies in $\mathcal{G}(\mathcal{A}, T)$ are in bijection with (accepting) runs of $\mathcal{A}$ on T. Moreover, the game $\mathcal{G}(\mathcal{A}, T)$ has the same winning strategies as the usual acceptance games (see e.g. [Tho97, GTW02, PP04]).*

Remark A.7 (Tree Games v.s. Graph Games)**.** *The realm of game semantics was originally mostly developed with games on trees, (such as simple games [Abr97, Hyl97], but also traditional Hyland-Ong games [HO00]). However, more recent trends of game semantics based on graphs also emerged, in particular in the work of Melliès (see e.g. [Mel05, Mel06, Mel12]). Wrt. the context of this paper, one should in particular note the connection of [Mel06] between innocence and a notion of positionality (but for games equipped with an asynchronous notion of monoidal product).*

On the other hand, the framework of [Rib15] was itself based on games on graphs, and games on graphs will be considered in this paper in relation with positionality in §7.2.3. However, that notion of positionality is not yet clearly connected with the notion of positionality of [Mel06], most notably because of the synchronous *nature of our games, and in particular of their monoidal structure (see §4.1). Moreover, the main developments of this paper, based on the category* **DZ** *of simple zig-zag games (see §3) are technically easier with games on trees, and we adopt this setting for the categories of games considered here.*

A.2. Game Semantics: Linear Arrow Games and Copy-Cat. We now present the usual notion of morphism of simple games (see e.g. [Abr97, Hyl97]), on which are based the morphisms of [Rib15] as well as those of the categories of the present paper. Our motivation to rely on this notion is to solve the problem of §2.3. Recall from §2.3 that a sequent $M \;;\; \mathcal{A} \vdash \mathcal{B}$ (for $\mathcal{A}, \mathcal{B}$ usual non-deterministic automata) should be thought as a form of implication, but that the runs of the automaton $\sim\mathcal{A} \oplus \mathcal{B}$ seemed not to convey the right information. The first encountered difficulty concerned the existence of *canonical* identities $\mathrm{id}_{\mathcal{A}(M)} \in \mathbb{E}_\Sigma[\mathcal{A}(M), \mathcal{A}(M)]$ if the homset $\mathbb{E}_\Sigma[\mathcal{A}(M), \mathcal{A}(M)]$ were to be the set of accepting runs or winning P-strategies $(\sim\mathcal{A})(M) + \mathcal{A}(M)$. The solution of *game semantics* is to devise, from component games A and B, an implication game $A \multimap_{\mathbf{SG}} B$ in which the games $\overline{A}$ and B are interleaved. More precisely:

(a) The set of moves of $A \multimap_{\mathbf{SG}} B$ is the disjoint union of the sets of moves of A and B, and the components A and B can be interleaved in plays on $A \multimap_{\mathbf{SG}} B$.

(b) O plays first in $A \multimap_{\mathbf{SG}} B$, and then the plays in $A \multimap_{\mathbf{SG}} B$ alternate between P and O.

(c) The role of P and O are reversed in component A and are preserved in component B (*i.e.* P in $A \multimap_{\mathbf{SG}} B$ plays as O in A and as P in B).

(d) In the case of *simple games*, P can switch between components at any of its moves, but O must stay in the same component (this is the *switching condition*).

Definition A.8 (Linear Arrow Games)**.** *Given polarized simple games A and B of the same* polarity*, the* linear arrow game *$A \multimap_{\mathbf{SG}} B$ is the negative game*

$$A \multimap_{\mathbf{SG}} B \quad := \quad (A_{\mathsf{O}} + B_{\mathsf{P}},\ A_{\mathsf{P}} + B_{\mathsf{O}},\ L_{A \multimap_{\mathbf{SG}} B})$$

where $L_{A \multimap_{\mathbf{SG}} B}$ consists of those negative plays s such that $s_{\restriction A} \in L_A$ and $s_{\restriction B} \in L_B$, where $s_{\restriction A}$ is the restriction of s to $A_{\mathsf{P}} + A_{\mathsf{O}}$, and similarly for $s_{\restriction B}$.

Let us check that $A \multimap_{\mathbf{SG}} B$ satisfies the *switching condition* (d) above (the other conditions (a)-(c) are direct consequences of the definitions): given a legal O-play $s = t \cdot n \cdot m$, either n, m are both in component A, or they are both in component B. Indeed, note that since $A \multimap_{\mathbf{SG}} B$ is negative, its legal O-plays are odd-length. So if s is a legal O-play, then the lengths of $s_{\restriction A}$ and $s_{\restriction B}$ cannot have the same parity. Assume now that $s = t \cdot n \cdot m$ with n and m in different components. Since A and B are assumed to be of the same polarity, the moves

	$A \xrightarrow{\mathrm{id}_A} \multimap_{\mathbf{SG}} A$		
	$\vdots$	$\vdots$	
O	u		
		u	P
		x	O
P	x		
		$\vdots$	

Figure 25: A play of the *copy-cat* identity strategy id

n and m are of different polarities w.r.t. A and B, so they are of the same polarity as moves of $A \multimap_{\mathbf{SG}} B$, contradicting the legality of s.

Simple games and (winning) strategies form a category $\mathbf{SG}^{(\mathrm{W})}$, whose objects are simple games (with winning), and whose morphisms from A to B are (winning) P-strategies $\sigma : A \multimap_{\mathbf{SG}} B$. We refer to [Abr97, Hyl97, AC98] for full treatments, and in particular to [Abr97, Hyl97] for totality and winning. The general notion of winning in games of the form $A \multimap_{\mathbf{SG}} B$ is a bit technical. In this paper, we only need to consider the case of infinite plays on $A \multimap_{\mathbf{SG}} B$ whose projections on A and B are both infinite. We say that such a play is winning for P in $A \multimap_{\mathbf{SG}} B$ iff its projection on B is winning for P whenever its projection on A is winning for P (with the original polarities of A).

Consider now the definition of the identity strategy id_A in $A \multimap_{\mathbf{SG}} A$ for $A = (U, X)$ a full positive game. Since O must begin in $A \multimap_{\mathbf{SG}} A$, but it is P who begins in the right component A, it follows that O must begin in the left component A (taking the role of P in that component). It is then easy to define an identity "*copy-cat*" strategy for P, which always switches component and copies the previous O-move from the other component. A play of this strategy is depicted in Fig. 25 (where plays grow from top to bottom). Formally, id_A is the unique strategy in $A \multimap_{\mathbf{SG}} A$ such that

$$\mathrm{id}_A \quad = \quad \{s \in L_{A^0 \multimap_{\mathbf{SG}} A^1} \mid s_{\restriction A^0} = s_{\restriction A^1}\} \tag{40}$$

(where we have written $A \multimap_{\mathbf{SG}} A$ as $A^0 \multimap_{\mathbf{SG}} A^1$ in order to distinguish the two copies of A).

In particular the same sequences of moves are produced by id_A in both copies of A. Assuming that A is equipped with a winning condition $\mathcal{W}_A$, such sequences are either winning for P in A or are winning for O in A. So they are winning for P in $A \multimap_{\mathbf{SG}} A$.

A.3. Uniform Linear Synchronous Arrow Games. We now present the notion of *linear synchronous arrow games* of [Rib15] adapted to uniform automata (Def. 3.1). Although we do not formally need this notion of games in this paper, it actually underlies the morphisms of the categories DialZ$(-)$ and DialAut of §4, and motivates our use of zig-zag games.

Synchronous arrow games formalize the constraint on plays of Fig. 4 (left) in §3.5. They are restrictions of the linear arrow of simple games between substituted acceptance games, in which P has to play the same input letters a and the same tree directions d as proposed by O.

Consider substituted acceptance games $\Sigma \vdash \mathcal{A}(M), \mathcal{B}(N)$ (in the sense of §3.3) with $\mathcal{A}$ as in (27) and

$$\mathcal{B} \quad = \quad (Q_{\mathcal{B}}\,,\, q^{\imath}_{\mathcal{B}}\,,\, V\,,\, Y\,,\, \partial_{\mathcal{B}}\,,\, \Omega_{\mathcal{B}})$$

We define the *synchronous arrow game*

$$\Sigma \vdash \mathcal{A}(M) \multimap \mathcal{B}(N) \tag{41}$$

also noted $\mathcal{A}(M) \multimap \mathcal{B}(N)$ when Σ is clear from the context.

The game $\mathcal{A}(M) \multimap \mathcal{B}(N)$ will be a subgame of $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$. It can be seen as a restriction of $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$ to plays which are *synchronous*, in the sense that $\mathcal{A}$ and $\mathcal{B}$ are evaluated along the same path in $\mathfrak{D}^\omega$, while M and N read the same input letters from Σ. The synchronous plays of $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$ are defined using the following notion of *trace*. Let

$$\mathrm{Tr}_\Sigma \quad := \quad (\Sigma \cdot \mathfrak{D})^* + (\Sigma \cdot \mathfrak{D})^* \cdot \Sigma$$

and define the *trace function* $\mathrm{tr}_{\mathcal{A}(M)} : L_{\mathcal{A}(M)} \longrightarrow \mathrm{Tr}_\Sigma$ inductively as follows

$$\begin{array}{rcl} \mathrm{tr}_{\mathcal{A}(M)}(\epsilon) & := & \epsilon \\ \mathrm{tr}_{\mathcal{A}(M)}(s \cdot (\mathtt{a}, u)) & := & \mathrm{tr}_{\mathcal{A}(M)}(s) \cdot \mathtt{a} \\ \mathrm{tr}_{\mathcal{A}(M)}(s \cdot (x, d)) & := & \mathrm{tr}_{\mathcal{A}(M)}(s) \cdot d \end{array}$$

We let the *trace* of a play $s \in L_{\mathcal{A}(M)}$ be the sequence $\mathrm{tr}_{\mathcal{A}(M)}(s)$. The trace function $\mathrm{tr}_{\mathcal{B}(N)} : L_{\mathcal{B}(N)} \longrightarrow \mathrm{Tr}_\Sigma$ is defined similarly. Note that both $\mathrm{tr}_{\mathcal{A}(M)}$ and $\mathrm{tr}_{\mathcal{B}(N)}$ have the same codomain Tr_Σ, which only depends on the input alphabet of M and N. Consider now a legal play s in $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$. We say that s is *synchronous* if

$$\mathrm{tr}_{\mathcal{A}(M)}(s_{\restriction \mathcal{A}(M)}) \quad = \quad \mathrm{tr}_{\mathcal{B}(N)}(s_{\restriction \mathcal{B}(N)})$$

Note that trace functions are length-preserving, so that the trace of a play s always has the same length as s. Hence if s is a synchronous play in $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$, then $s_{\restriction A(M)}$ and $s_{\restriction B(N)}$ have the same length, so that s is even length. It follows that the synchronous plays of $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$ must be P-plays.

We define the game $\mathcal{A}(M) \multimap \mathcal{B}(N)$ as the game $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$, but with as legal plays the legal plays of $\mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$ which are prefixes of synchronous plays. It follows that a P-strategy $\sigma : \mathcal{A}(M) \multimap \mathcal{B}(N)$ is a P-strategy $\sigma : \mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{B}(N)$ whose plays are all synchronous. We call such strategies *synchronous*. In particular, the identity copy-cat strategy $\mathrm{id}_{\mathcal{A}(M)} : \mathcal{A}(M) \multimap_{\mathbf{SG}} \mathcal{A}(M)$ is a synchronous strategy.

A typical synchronous play in $\mathcal{A}(M) \multimap \mathcal{B}(N)$ is depicted in Fig. 26. Note that synchronous plays must have the same zig-zag shape as the copy-cat plays, and moreover that O actually chooses both the input letters $\mathtt{a} \in \Sigma$ and the tree directions $d \in \mathfrak{D}$. This follows from the fact that in the game $\mathcal{A}(M) \multimap \mathcal{B}(N)$, O must begin in the component $\mathcal{A}(M)$, choosing in particular some $\mathtt{a} \in \Sigma$. Then, by synchronicity, P must switch to component $\mathcal{B}(N)$ and play a move containing *the same* $\mathtt{a} \in \Sigma$. Since O cannot switch component, its next move must be in component $\mathcal{B}(N)$, and so in particular contain some $d \in \mathfrak{D}$. But then, again by synchronicity, P must switch to component $\mathcal{A}(M)$ and play a move containing the same $d \in \mathfrak{D}$.

Synchronous arrow games can be equipped with the winning conditions mentioned in §A.2. Given an infinite play χ in $\mathcal{A}(M) \multimap \mathcal{B}(N)$ whose projections on $\mathcal{A}(M)$ and $\mathcal{B}(N)$ are both infinite, we say that χ is *winning for* P if its projection on $\mathcal{B}(N)$ is winning for P whenever its projection on $\mathcal{A}(M)$ is winning for P.

A.4. Zig-Zag Games. We now discuss the zig-zag games of §3.4 in the context of simple games, and explain how they arise from the synchronous arrow games presented in §A.3.

Consider substituted acceptance games $\Sigma \vdash \mathcal{A}(M), \mathcal{B}(N)$ as in §A.3. Recall that the synchronicity constraint of §A.3 imposes a legal P-play s in $\mathcal{A}(M) \multimap \mathcal{B}(N)$ to satisfy

$$\mathrm{tr}_{\mathcal{A}(M)}(s_{\restriction \mathcal{A}(M)}) \quad = \quad \mathrm{tr}_{\mathcal{B}(N)}(s_{\restriction \mathcal{B}(N)})$$

Σ	$\mathcal{A}(M)$	$\multimap$	$\mathcal{B}(N)$	
	$(\epsilon, \epsilon, q^{\iota}_{\mathcal{A}})$		$(\epsilon, \epsilon, q^{\iota}_{\mathcal{B}})$	
	$\vdots$		$\vdots$	
	$(p, \overline{\mathsf{a}}, q_{\mathcal{A}})$		$(p, \overline{\mathsf{a}}, q_{\mathcal{B}})$	
O	(a, u)			
			(a, v)	P
			(y, d)	O
P	$x, d)$			
	$(p \cdot d,\ \overline{\mathsf{a}} \cdot \mathsf{a},\ q'_{\mathcal{A}})$		$(p \cdot d,\ \overline{\mathsf{a}} \cdot \mathsf{a},\ q'_{\mathcal{B}})$	
	$\vdots$		$\vdots$	

Figure 26: A typical synchronous play over Σ

	A	$\multimap_{\mathbf{DZ}}$	B	
	$\vdots$		$\vdots$	
O	u			
			v	P
			y	O
P	x			
	$\vdots$		$\vdots$	

Figure 27: A typical zig-zag play with $A = (U, X)$ and $B = (V, Y)$ full positive games

Since the functions $\mathrm{tr}_{\mathcal{A}(M)}$ and $\mathrm{tr}_{\mathcal{B}(N)}$ are length-preserving, this imposes in particular $s_{\restriction \mathcal{A}(M)}$ and $s_{\restriction \mathcal{B}(N)}$ to have the same length. On the other hand, given simple games A and B of the same polarity, and a play s in $A \multimap_{\mathbf{SG}} B$, if

$$\mathrm{length}(s_{\restriction A}) \quad = \quad \mathrm{length}(s_{\restriction B}) \tag{42}$$

then in s, each P-move must switch component w.r.t. the previous O-move. Let us discuss the case where (say) $A = (U, X)$ and $B = (V, Y)$ are full positive games. Recall that O begins in $A \multimap_{\mathbf{SG}} B$ and must play in component A since A and B are positive. In order to maintain (42), P must then switch to component B. After the P-move in B, the switching condition imposes O to stay in B, and then P has to switch to A, again to maintain (42). It follows that s must have the *zig-zag* shape depicted in Fig. 27.

This leads to the following notion of zig-zag strategies in the setting of simple games.

Definition A.9 (Zig-Zag Plays and Strategies)**.** *Given simple games A and B of the same polarity, a play s in $A \multimap_{\mathbf{SG}} B$ is a* zig-zag play *if*

$$\mathrm{length}(s_{\restriction A}) \quad = \quad \mathrm{length}(s_{\restriction B})$$

A P*-strategy $\sigma : A \multimap_{\mathbf{SG}} B$ is a* zig-zag *strategy if all its plays are zig-zag plays.*

We write $A \multimap_{\mathbf{DZ}} B$ for the game obtained by restricting $A \multimap_{\mathbf{SG}} B$ to (prefixes of) its legal zig-zag plays (so the P*-strategies on $A \multimap_{\mathbf{DZ}} B$ are exactly the zig-zag* P*-strategies on $A \multimap_{\mathbf{SG}} B$).*

Consider now games with winning A and B. Note that if $\sigma : A \multimap_{\mathbf{DZ}} B$ is total, then for every $\chi \in ((A_{\mathsf{P}} + B_{\mathsf{O}}) \cdot (A_{\mathsf{O}} + B_{\mathsf{P}}))^\omega$, if χ has infinitely many finite prefixes in σ, then $\chi_{\restriction A}$ and $\chi_{\restriction B}$ are both infinite. We therefore let $\mathcal{W}_{A \multimap B} \subseteq ((A_{\mathsf{P}} + B_{\mathsf{O}}) \cdot (A_{\mathsf{O}} + B_{\mathsf{P}}))^\omega$ be the set of infinite sequences χ such that $(\chi_{\restriction A} \in \mathcal{W}_A \implies \chi_{\restriction B} \in \mathcal{W}_B)$.

For full positives games, total zig-zag strategies in the sense of Def. A.9 correspond exactly to those of Def. 3.9.

Proposition A.10. *Consider full positive games $A = (U, X)$ and $B = (V, Y)$.* Total *zig-zag strategies $\sigma : A \multimap_{\mathbf{DZ}} B$ are in bijection with pairs of functions (f, F) where*

$$\begin{array}{rcccl} f & : & \bigcup_{n\in\mathbb{N}} \left(U^{n+1} \times Y^n\right) & \longrightarrow & V \\ F & : & \bigcup_{n\in\mathbb{N}} \left(U^{n+1} \times Y^{n+1}\right) & \longrightarrow & X \end{array} \tag{43}$$

Given pairs of maps (f, F) as in (43), for each $n > 0$, we write f_n and F_n for the induced maps

$$f_n \ : \ U^n \times Y^{n-1} \ \longrightarrow \ V^n \qquad \text{and} \qquad F_n \ : \ U^n \times Y^n \ \longrightarrow \ X^n$$

Note that for each full positive game $A = (U, X)$, there is a bijection (were $\wp_A^{\mathsf{even}}$ denotes the set of even-length plays of A):

$$\partial \ = \ \langle \partial_U, \partial_X \rangle \quad : \quad \wp_A^{\mathsf{even}} \quad \longrightarrow \quad \cup_{n\in\mathbb{N}}(U^n \times X^n)$$

defined as $\partial(\epsilon) := (\epsilon, \epsilon)$ and $\partial(s.u.x) = (\partial_U(s).u, \partial_X(s).x)$.

Proof of Prop. A.10. Fix $A = (U, X)$ and $B = (V, Y)$ and consider a total zig-zag strategy $\sigma : A \multimap_{\mathbf{DZ}} B$. By induction on $n \in \mathbb{N}$, it is easy to see that for all $(\overline{u}, \overline{y}) \in U^n \times Y^n$, there is a unique $(s, t) \in \mathrm{HS}(\sigma)$ such that $\overline{u} = \partial_U(s)$ and $\overline{y} = \partial_Y(t)$. The property vacuously holds for $n = 0$. Assuming it for n, given $(\overline{u}.u, \overline{y}.y) \in U^{n+1} \times Y^{n+1}$, by induction hypothesis, there is a unique $(s, t) \in \mathrm{HS}(\sigma)$ such that $\overline{u} = \partial_U(s)$ and $\overline{y} = \partial_Y(t)$. Now, since σ is total and zig-zag, there is a unique $v \in V$ such that $(s.u, t.v) \in \mathrm{HS}(\sigma)$. Similarly, there is a unique $x \in X$ such that $(s.u.x, t.v.y) \in \mathrm{HS}(\sigma)$, and the property follows. Furthermore, since $\overline{u}.u$ and $\overline{y}$ uniquely determine $\overline{v} = \partial_V(t)$ and v, and since $\overline{u}.u$ and $\overline{y}.y$ uniquely determine $\overline{x} = \partial_X(s)$ and x, we obtain a pair of functions (f, F) as in (43) defined as

$$f(\overline{u}.u, \overline{y}) \ := \ v \qquad \text{and} \qquad F(\overline{y}.y, \overline{u}.u) \ := \ x$$

Conversely, each pair (f, F) as in (43) uniquely determines a total zig-zag strategy σ, with, for all $\overline{u}.u \in U^{n+1}$, and all $\overline{y} \in Y^n$,

$$(\partial^{-1}(\overline{u}, \overline{x}).u \ , \ \partial^{-1}(\overline{v}, \overline{y}).v) \ \in \ \mathrm{HS}(\sigma)$$

where $\overline{v}.v = f_{n+1}(\overline{u}.u, \overline{y})$ and $\overline{x} = F_n(\overline{u}, \overline{y})$; and moreover for all y,

$$(\partial^{-1}(\overline{u}, \overline{x}).u.x \ , \ \partial^{-1}(\overline{v}, \overline{y}).v.y) \ \in \ \mathrm{HS}(\sigma)$$

where $x = F(\overline{u}.u, \overline{y}.y)$. □

A.5. The Category $\mathbf{DZ}^{(\mathrm{W})}$ of Zig-Zag Games and Total (Winning) Strategies. We now discuss composition and identities for total (winning) zig-zag strategies. This will lead us to prove Prop. 3.10, namely that $\mathbf{DZ}$ and $\mathbf{DZ}^{\mathrm{W}}$ are categories.

Let us begin by informally discussing composition of zig-zag strategies as presented in Def. A.9, and following the usual intuitions of game semantics. We refer to e.g. [Abr97, Hyl97] for a description of composition in $\mathbf{SG}$, and to [Rib15] for the particular case of synchronous strategies. Consider full positive games $A = (U, X)$, $B = (V, Y)$ and $C = (W, Z)$. Consider also P-strategies $\sigma : A \multimap_{\mathbf{DZ}} B$ and $\tau : B \multimap_{\mathbf{DZ}} C$ as in Fig. 28 (top). Their composite

$$\tau \circ \sigma \quad : \quad A \quad \longrightarrow_{\mathbf{DZ}} \quad C$$

is obtained by making σ and τ interact in their common component B, as depicted on Fig. 28 (middle). The crucial observation is that in an interaction of σ and τ in component B, all the P-moves are played by σ and all the O-moves are played by τ. It follows that the interactions of σ and τ in component B are completely determined by σ and τ and the O-moves in $A \multimap_{\mathbf{DZ}} C$. The composite strategy $\tau \circ \sigma$ is then obtained by hiding the interaction of σ and τ in their common component B (see Fig. 28, bottom).

We now come to the formal treatment of composition and identities in $\mathbf{DZ}^{(\mathrm{W})}$. Let us step back to some well-known facts on simple games from [HS99]. There is a faithful functor $\mathrm{HS} : \mathbf{SG} \longrightarrow \mathbf{Rel}$ (the category of sets and relations) taking a simple game A to its set of legal plays L_A, and a strategy $\sigma : A \multimap_{\mathbf{SG}} B$ to

$$\mathrm{HS}(\sigma) \quad := \quad \{(s_{\restriction A}, s_{\restriction B}) \mid s \in \sigma\} \quad \subseteq \quad L_A \times L_B$$

We can therefore faithfully represent strategies $\sigma : A \multimap B$ as spans

$$\begin{tikzcd} & \mathrm{HS}(\sigma) \arrow[dl, "\pi_1"'] \arrow[dr, "\pi_2"] & \\ L_A & & L_B \end{tikzcd} \tag{44}$$

and moreover, composition and identities of simple game can be seen as being induced from composition and identities on relations. Explicitly $\mathrm{HS}(\mathrm{id}_A)$ is identity relation on L_A as suggested by (40)), and given strategies $\sigma : A \multimap_{\mathbf{SG}} B$ and $\tau : B \multimap_{\mathbf{SG}} C$, the strategy $\tau \circ \sigma$ is the unique strategy such that $\mathrm{HS}(\tau \circ \sigma)$ is the relation $\mathrm{HS}(\tau) \circ \mathrm{HS}(\sigma)$.

In other words, the category $\mathbf{SG}$ of simple games can be obtained from the category $\mathbf{Rel}$ thanks to the injectivity of HS seen as function from strategies to relations. In the case of total zig-zag strategies, composition and identities can be obtained along this scheme, but with much simpler combinatorics than with $\mathbf{SG}$.

First, note that the map (that we still denote HS)

$$\mathrm{HS} \quad : \quad s \in L_{A \multimap_{\mathbf{SG}} B} \quad \longmapsto \quad (s_{\restriction A}, s_{\restriction B}) \in L_A \times L_B$$

is injective on *zig-zag* plays: given $(t, t') \in L_A \times L_B$, there is at most one *zig-zag* play s such that $\mathrm{HS}(s) = (t, t')$. This immediately gives the injectivity of HS on *zig-zag* strategies.

Lemma A.11. *(i) Given* zig-zag *plays* s, t *in* $A \multimap_{\mathbf{DZ}} B$*, if* $\mathrm{HS}(s) = \mathrm{HS}(t)$ *then* $s = t$*.*

(ii) The map HS *is injective on* zig-zag *strategies:* $\mathrm{HS}(\sigma) = \mathrm{HS}(\tau)$ *implies* $\sigma = \tau$*.*

Second, the representation of strategies as pairs of maps (f, F) of the form (43) provides an easy way to compose total zig-zag strategies. Given total zig-zag strategies $\sigma : A \multimap_{\mathbf{DZ}} B$ and

	A	$\xrightarrow{\sigma}$	B	
	$\vdots$		$\vdots$	
O	u			
			v	P
			y	O
P	x			
	$\vdots$		$\vdots$	

	B	$\xrightarrow{\tau}$	C	
	$\vdots$		$\vdots$	
O	v			
			w	P
			z	O
P	y			
	$\vdots$		$\vdots$	

	A	$\xrightarrow{\sigma}$	B	$\xrightarrow{\tau}$	C	
	$\vdots$		$\vdots$		$\vdots$	
O	u					O
			v			
P					w	P
O					z	O
			y			
P	x					P
	$\vdots$		$\vdots$		$\vdots$	

	A	$\xrightarrow{\tau\circ\sigma}$	C	
	$\vdots$		$\vdots$	
O	u			
			w	P
			z	O
P	x			
	$\vdots$		$\vdots$	

Figure 28: Interaction of zig-zag strategies on full positive games

$\tau : B \multimap_{\mathbf{DZ}} C$, we are looking for a composite $\tau \circ \sigma$. By injectivity of HS, it is sufficient to show that there exists a strategy θ such that $\mathrm{HS}(\theta) = \mathrm{HS}(\tau) \circ \mathrm{HS}(\sigma)$. But thanks to Prop. A.10, given pairs of maps (f, F) and (g, G) representing resp. σ and τ, this amounts to provide a pair (h, H) representing θ. Write $A = (U, X)$, $B = (V, Y)$ and $C = (W, Z)$. The relational composite $\mathrm{HS}(\tau) \circ \mathrm{HS}(\sigma)$ is such that $(\partial^{-1}(\overline{u}, \overline{x}), \partial^{-1}(\overline{w}, \overline{z})) \in \mathrm{HS}(\tau) \circ \mathrm{HS}(\sigma)$ if and only if there are $(\overline{v}, \overline{y})$ such that

$$(\partial^{-1}(\overline{u}, \overline{x}) \,,\, \partial^{-1}(\overline{v}, \overline{y})) \in \mathrm{HS}(\sigma) \qquad \text{and} \qquad (\partial^{-1}(\overline{v}, \overline{y}) \,,\, \partial^{-1}(\overline{w}, \overline{z})) \in \mathrm{HS}(\tau)$$

But by Prop. A.10 this is possible if and only if the following equations are satisfied:

$$\begin{array}{rclcrcl} \overline{v} & = & f_n(\overline{u}, \blacktriangleright\overline{y}) & \qquad & \overline{w} & = & g_n(\overline{v}, \blacktriangleright\overline{z}) \\ \overline{x} & = & F_n(\overline{u}, \overline{y}) & & \overline{y} & = & G_n(\overline{v}, \overline{z}) \end{array} \tag{45}$$

(where $\blacktriangleright\epsilon := \epsilon$ and $\blacktriangleright\overline{y}.y := \overline{y}$). The derived equation

$$\overline{y} \quad = \quad G_n(f_n(\overline{u}, \blacktriangleright\overline{y}) \,,\, \overline{z}) \tag{46}$$

determines $\overline{y} = y(\overline{u}, \overline{z}) = y_1 \ldots . y_n$ uniquely from $\overline{u} = u_1 \ldots . u_n$ and $\overline{z} = z_1 \ldots . z_n$, as

$$y_k \;=\; G(f_k(u_1 \ldots . u_k, \, y_1 \ldots . y_{k-1}) \,,\, z_1 \ldots . z_k) \tag{47}$$

We can thus define a pair of maps

$$\begin{array}{rcccl} h & : & \bigcup_{n \in \mathbb{N}} \left(U^{n+1} \times Z^n\right) & \longrightarrow & W \\ H & : & \bigcup_{n \in \mathbb{N}} \left(U^{n+1} \times Z^{n+1}\right) & \longrightarrow & X \end{array}$$

as follows

$$\begin{array}{rcl} h(\overline{u}u, \, \overline{z}) & := & g(f_{n+1}(\overline{u}u, \, y(\overline{u}, \overline{z}))) \,,\, \overline{z}) \\ H(\overline{u}u, \, \overline{z}z) & := & F(\overline{u}u, \, y(\overline{u}u, \overline{z}z)) \end{array}$$

Then, by construction of (h, H), the total strategy $\theta : A \multimap_{\mathbf{DZ}} C$ it represents is such that $\mathrm{HS}(\theta) = \mathrm{HS}(\tau) \circ \mathrm{HS}(\sigma)$, so that we can let $\tau \circ \sigma := \theta$.

Note that the strategy $\tau \circ \sigma$ is total. Hence, totality is preserved by composition of *zig-zag* strategies, while on the other hand, it is well-known that totality is *not* preserved by composition of *arbitrary* **SG**-strategies (see e.g. [Abr97]).

We alluded to the usual method to compose **SG**-strategies $\sigma : A \multimap_{\mathbf{SG}} B$ and $\tau : B \multimap_{\mathbf{SG}} C$, which proceeds by letting σ and τ interact in their common component B, and then hiding this interaction (see e.g. [Abr97, Hyl97] for details). This relies on the usual *zipping* property, stating that the interactions of σ and τ in component B are completely determined by the O-moves in components A and C (with the polarities of $A \multimap_{\mathbf{SG}} C$). In our case, the zipping property follows from the definitions of $\overline{y}$ from $\overline{u}$ and $\overline{z}$ by (47), and of $\overline{v}$ and $\overline{x}$ from $\overline{u}$ and $\overline{y}$ in (45). We have in particular the following relational version of zipping (which actually holds for the full **SG** [HS99]):

Lemma A.12 (Relational Zipping)**.** *Given total zig-zag $\sigma : A \multimap_{\mathbf{DZ}} B$ and $\tau : B \multimap_{\mathbf{DZ}} C$, and given $(t_A, t_C) \in \mathrm{HS}(\tau) \circ \mathrm{HS}(\sigma)$, there is exactly one legal play $t_B \in L_B$ such that $(t_A, t_B) \in \mathrm{HS}(\sigma)$ and $(t_B, t_C) \in \mathrm{HS}(\tau)$.*

From the relational zipping Lemma A.12 one gets the usual and expected fact that if $\sigma : A \multimap_{\mathbf{DZ}} B$ and $\tau : B \multimap_{\mathbf{DZ}} C$ are both total and winning, then $\tau \circ \sigma$ is total and winning. Indeed, given an infinite play $\chi \in ((A_{\mathsf{O}} + C_{\mathsf{P}}) \cdot (A_{\mathsf{P}} + C_{\mathsf{O}}))^\omega$ of $\tau \circ \sigma$ (that is such that $\exists^\infty k \in$

$\mathbb{N}$. $\chi(0) \cdot \ldots \cdot \chi(k) \in \tau \circ \sigma$), it follows from Lem. A.12 that there are infinite plays χ_σ and χ_τ of resp. σ and τ such that

$$(\chi_\sigma)_{\restriction A} \quad = \quad \chi_{\restriction A} \qquad \text{and} \qquad (\chi_\sigma)_{\restriction B} \quad = \quad (\chi_\tau)_{\restriction B} \qquad \text{and} \qquad (\chi_\tau)_{\restriction C} \quad = \quad \chi_{\restriction C}$$

from which we get

$$(\chi_{\restriction A} \in \mathcal{W}_A) \quad \Longrightarrow \quad ((\chi_\sigma)_{\restriction B} \;=\; (\chi_\tau)_{\restriction B} \;\in\; \mathcal{W}_B) \quad \Longrightarrow \quad (\chi_{\restriction C} \in \mathcal{W}_C)$$

We thus have

Proposition A.13 (Prop. 3.10)**.** *Full positive games (with winning) and (winning) total zig-zag strategies form a category* $\mathbf{DZ}^{(\mathrm{W})}$*.*

Note that the identity and associativity laws for composition of strategies are lifted from the corresponding laws in **Rel** by the injectivity of HS. On the one hand, we have

$$\mathrm{id}_B \circ \sigma \quad = \quad \sigma \quad = \quad \sigma \circ \mathrm{id}_A$$

since $\mathrm{HS}(\mathrm{id}_A)$ (resp. $\mathrm{HS}(\mathrm{id}_B)$)) is the identity relation on L_A (resp. L_B) and since

$$\mathrm{HS}(\mathrm{id}_B) \circ \mathrm{HS}(\sigma) \quad = \quad \mathrm{HS}(\sigma) \quad = \quad \mathrm{HS}(\sigma) \circ \mathrm{HS}(\mathrm{id}_A)$$

On the other hand, the associativity of composition (that is $\theta \circ (\tau \circ \sigma) = (\theta \circ \tau) \circ \sigma$) follows from the fact that $\mathrm{HS}(\theta) \circ (\mathrm{HS}(\tau) \circ \mathrm{HS}(\sigma)) = (\mathrm{HS}(\theta) \circ \mathrm{HS}(\tau)) \circ \mathrm{HS}(\sigma)$.

B. Proof of Adequacy of the Promotion Rule (Prop. 7.13)

We give here a detailed proof of Prop. 7.13. The argument is essentially the same as that of [Wal02], with the obvious adaptations to our (slightly more complicated) setting.

Proposition B.1 (Prop. 7.13)**.** *Given* $\mathcal{N}, \mathcal{A} : \Sigma$ *with* $\mathcal{N}$ *non-deterministic, if there is a winning* P*-strategy in* $\Sigma \vdash \mathcal{N}(L) \multimap \mathcal{A}(M)$ *then there is a winning* P*-strategy in* $\Sigma \vdash \mathcal{N}(L) \multimap !\mathcal{A}(M)$*.*

Proof. Write $I = \{*\}$ for the set of O-moves of $\mathcal{N}$. By Prop. 7.10, we can assume $\mathcal{N}$ and $\mathcal{A}$ to be parity automata. Write G for the game graph of $\Sigma \vdash \mathcal{N}(L) \multimap \mathcal{A}(M)$. Thanks to [Kla94, KK95, Jut97, Zie98], there is a positional (w.r.t. G) winning P-strategy σ in $\Sigma \vdash \mathcal{N}(L) \multimap \mathcal{A}(M)$.

We build a winning P-strategy τ on $\mathcal{N}(L) \multimap !\mathcal{A}(M)$ such that the following invariant is satisfied:

- to each play t of τ with $\mathsf{pos}(t) = ((p, \overline{\mathsf{a}}, q_{\mathcal{N}}), (p, \overline{\mathsf{a}}, S))$ and $\pi_2(S) = \{q_1, \ldots, q_n\}$, we associate a set $E(t) = \{s_1, \ldots, s_n\}$ of plays of σ, with $\mathsf{pos}(s_i) = ((p, \overline{\mathsf{a}}, q_{\mathcal{N}}), (p, \overline{\mathsf{a}}, q_i))$ for each $1 \leq i \leq n$.
- and if moreover t' extends t and is such that $\mathsf{pos}(t') = ((p.d, \overline{\mathsf{a}}.\mathsf{a}, q'_{\mathcal{N}}), (p.d, \overline{\mathsf{a}}.\mathsf{a}, S'))$ then for all $s' \in E(t')$ there is some $s \in E(t)$ such that s' extends s.

The strategy τ is built by induction on plays as follows:

- For the base case (initial position ϵ), we have by definition $S = \{(q^\imath_{\mathcal{A}}, q^\imath_{\mathcal{A}})\}$ and $E(\epsilon) = \{q^\imath_{\mathcal{A}}\}$.

- For the inductive step, let t with $\mathsf{pos}(t) = ((p, \overline{\mathsf{a}}, q_{\mathcal{N}})\,,\,(p, \overline{\mathsf{a}}, S))$ and let O play from t some (a, v) in component $\mathcal{N}(L)$ of $\mathcal{N}(L) \multimap !\mathcal{A}$.

  For $s_i \in E(t)$, let u_i be the move of σ from position $((p, \overline{\mathsf{a}}.\mathsf{a}, q_{\mathcal{N}}, v)\,,\,(p, \overline{\mathsf{a}}, q_i))$, thus going to position $((p, \overline{\mathsf{a}}.\mathsf{a}, q_{\mathcal{N}}, v)\,,\,(p, \overline{\mathsf{a}}.\mathsf{a}, q_i, u_i))$. This defines a function

$$\begin{array}{rcccl} h_{t.(\mathsf{a},v)} & : & Q_{\mathcal{A}} & \longrightarrow & U \\ & & q_i & \longmapsto & u_i \end{array}$$

  (the value of $h_{t.(\mathsf{a},v)}$ on irrelevant q's is arbitrary). We then let τ play $h_{t.(\mathsf{a},v)}$ in the component $!\mathcal{A}(M)$ of $\mathcal{N}(L) \multimap !\mathcal{A}(M)$, thus going to position

$$((p, \overline{\mathsf{a}}.\mathsf{a}, q_{\mathcal{N}}, v)\,,\,(p, \overline{\mathsf{a}}.\mathsf{a}, S, h_{t.(\mathsf{a},v)}))$$

  Then O answers some $d \in \mathfrak{D}$ in the component $!\mathcal{A}(M)$, and we let P play $*$ in the component $\mathcal{N}(L)$. The current position in $\mathcal{N}(L) \multimap !\mathcal{A}(M)$ becomes

$$((p.d, \overline{\mathsf{a}}.\mathsf{a}, q'_{\mathcal{N}})\,,\,(p.d, \overline{\mathsf{a}}.\mathsf{a}, S'))$$

  where

$$\begin{array}{rrcl} & q'_{\mathcal{N}} & := & \partial_{\mathcal{N}}(q_{\mathcal{N}}, L(\overline{\mathsf{a}}.\mathsf{a}, p), v, *, d) \\ \text{and} & S' & := & \partial_{!\mathcal{A}}(S, M(\overline{\mathsf{a}}.\mathsf{a}, p), h_{t.(\mathsf{a},v)}, \bullet, d) \end{array}$$

  Let

$$t' \quad := \quad t \cdot (\mathsf{a}, v) \cdot h_{t.(\mathsf{a},v)} \cdot d \cdot *$$

  and write $\pi_2(S') = \{q'_1, \ldots, q'_m\}$. By definition of the transition function of $!\mathcal{A}$, each q'_j is equal to $\partial_{\mathcal{A}}(q_{i_j}, M(\overline{\mathsf{a}}.\mathsf{a}, p), u_{i_j}, x_j, d)$ for some i_j and some x_j (note that there might be several such i_j and x_j, but we select one). For each j, we let O play (x_j, d) in the component $\mathcal{A}(M)$ of $\mathcal{N}(L) \multimap \mathcal{A}(M)$ from position $((p, \overline{\mathsf{a}}.\mathsf{a}, q_{\mathcal{N}}, v)\,,\,(p, \overline{\mathsf{a}}.\mathsf{a}, q_{i_j}, u_{i_j}))$ thus going to position $((p, \overline{\mathsf{a}}.\mathsf{a}, q_{\mathcal{N}}, v)\,,\,(p.d, \overline{\mathsf{a}}.\mathsf{a}, q'_j))$. We then let P answer $*$ in the component $\mathcal{N}(L)$, thus leading to position

$$((p.d, \overline{\mathsf{a}}.\mathsf{a}, q'_{\mathcal{N}})\,,\,(p.d, \overline{\mathsf{a}}.\mathsf{a}, q'_j))$$

  We finally put

$$E(t') \quad := \quad \{s_{i_0}.(\mathsf{a}, v).u_{i_0}.(x_0, d).\bullet\ ,\ \cdots\ ,\ s_{i_m}.(\mathsf{a}, v).u_{i_m}.(x_m, d).\bullet\}$$

This completes the definition of τ.

We now show that τ is winning. Consider an infinite play $(t_n)_{n\in\mathbb{N}}$ of τ, and let $(q_n, S_n)_{n\in\mathbb{N}}$ be the associated sequence of states in $(Q_{\mathcal{N}} \times Q_{!\mathcal{A}})^\omega$. Assume that we have $(q_n)_n \in \Omega_{\mathcal{N}}$. We show that $(S_n)_n \in \Omega_{!\mathcal{A}}$. Let $(q'_n)_n$ be a trace in $(S_n)_n$, so that $(q'_n, q'_{n+1}) \in S_{n+1}$. We have to show that $(q'_n)_n \in \Omega_{\mathcal{A}}$. To this end, we show that $(q'_n)_n$ is generated by the projection on $\mathcal{A}(M)$ of an infinite play of σ.

Note that for all $n \in \mathbb{N}$,

$$\mathsf{pos}(t_{4n}) \quad = \quad ((p_n, \overline{\mathsf{a}}_n, q_n)\,,\,(p_n, \overline{\mathsf{a}}_n, S_n))$$

By construction, for each $n \in \mathbb{N}$ there are $s_n \in E(t_{4n})$ and $s'_n \in E(t_{4(n+1)})$, such that

$$\mathsf{pos}(s_n) \quad = \quad ((p_n, \overline{\mathsf{a}}_n, q_n)\,,\,(p_n, \overline{\mathsf{a}}_n, q'_n))$$

and such that s'_n extends s_n:

$$s'_n \quad = \quad s_n \cdot (\mathsf{a}_n, v_n) \cdot u_n \cdot d_n \cdot *$$

and such that moreover

$$\mathsf{pos}(s'_n) \quad = \quad ((p_{n+1}, \overline{\mathsf{a}}_{n+1}, q_{n+1}), (p_{n+1}, \overline{\mathsf{a}}_{n+1}, q'_{n+1}))$$

where $\overline{\mathsf{a}}_{n+1} = \overline{\mathsf{a}}_n.\mathsf{a}_n$ and $p_{n+1} = p_n.d_n$. Note that $\mathsf{pos}(s_n)$ is completely determined from $p_n, \overline{\mathsf{a}}_n$, which are induced by $(t_n)_n$, together with the states q_n and q'_n. It follows that for all $n \in \mathbb{N}$ we have

$$\mathsf{pos}(s'_n) \quad = \quad \mathsf{pos}(s_{n+1})$$

Since σ is positional, it follows that the infinite sequence

$$\chi \quad := \quad \epsilon.(\mathsf{a}_0, v_0).u_0.d_0.\cdots.p_n.(\mathsf{a}_n, v_n).u_n.d_n.\cdots$$

is an infinite play of σ. Since χ produces the sequence of states $(q_n, q'_n)_n \in (Q_{\mathcal{N}} \times Q_{\mathcal{A}})^\omega$, we get $(q'_n)_n \in \Omega_{\mathcal{A}}$ since $(q_n)_n \in \Omega_{\mathcal{N}}$ by assumption. □

C. Further Examples

This Appendix is devoted to detailed accounts of two known results on non-deterministic automata, which can be reformulated in our setting. The first result is the following uniform formulation of [CL08, Thm. 1].

Proposition C.1. *For each regular language $\mathcal{L} \subseteq \Sigma^{\mathfrak{D}^*}$, there is a non-deterministic automaton $\mathcal{B}$ with $\mathcal{L}(\mathcal{B}) = \mathcal{L}$, and such that for every non-deterministic parity automaton $\mathcal{A}$ with $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}$, there is a winning P-strategy in $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$ induced by a function $g : Q_{\mathcal{A}} \times Q_{\mathcal{B}} \times \Sigma \times U \to V$, where $\mathcal{A}$ (resp. $\mathcal{B}$) has set of P-moves U (resp. V).*

Our proof of Prop. C.1 relies on the existence of positional winning P-strategies in games of the form $\Sigma \vdash \mathcal{A} \otimes \mathcal{B} \multimap \perp$, for non-deterministic parity automata $\mathcal{A}, \mathcal{B} : \Sigma$ such that $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B}) = \emptyset$. Second, we show in §C.4 that such strategies, when combined with our internalized linear implication, can handle a construction for the separation property of [SA05, Thm. 2.7].

C.1. On Positional Strategies. Consider non-deterministic parity automata $\mathcal{A}, \mathcal{B} : \Sigma$. It follows from §7.2.3 that if P has a winning strategy in $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$, then P has a positional winning strategy. But the game graph of $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$ is equivalent to the graph $\widetilde{G}$ with vertices:

$$(A_{\mathsf{P}} \times B_{\mathsf{P}}) + (A_{\mathsf{O}} \times B_{\mathsf{P}}) + (A_{\mathsf{O}} \times B_{\mathsf{O}}) + (A_{\mathsf{O}} \times (B_{\mathsf{P}} \times \mathfrak{D}))$$

where

$$A_{\mathsf{P}} := Q_{\mathcal{A}} \qquad A_{\mathsf{O}} := \Sigma \times Q_{\mathcal{A}} \times U \qquad B_{\mathsf{P}} := Q_{\mathcal{B}} \qquad B_{\mathsf{O}} := \Sigma \times Q_{\mathcal{B}} \times V$$

and with edges depicted on Fig. 29.

Since a positional P-strategy in $\widetilde{G}$ is given by a function

$$g \quad : \quad Q_{\mathcal{A}} \times Q_{\mathcal{B}} \times \Sigma \times U \quad \longrightarrow \quad V$$

we thus have:

Lemma C.2. *Given non-deterministic parity automata $\mathcal{A}, \mathcal{B} : \Sigma$, if P has a winning strategy in $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$, then P has a winning strategy induced by a function $Q_{\mathcal{A}} \times Q_{\mathcal{B}} \times \Sigma \times U \to V$.*

$$
\begin{array}{c|ccc|l|l}
 & (q_{\mathcal{A}} & , & q_{\mathcal{B}}) & A_{\mathsf{P}} \times B_{\mathsf{P}} & \\
\mathsf{O} & \downarrow & & & & \\
 & ((\mathtt{a}\,,\, q_{\mathcal{A}}\,,\, u) & , & q_{\mathcal{B}}) & A_{\mathsf{O}} \times B_{\mathsf{P}} & \\
\mathsf{P} & & & \downarrow & & \\
 & (\mathtt{a}\,,\, q_{\mathcal{A}}\,,\, u) & , & (\mathtt{a}\,,\, q_{\mathcal{B}}\,,\, v)) & A_{\mathsf{O}} \times B_{\mathsf{O}} & \\
\mathsf{O} & & & \downarrow & & \text{where} \quad q'_{\mathcal{B}} := \partial_{\mathcal{B}}(q_{\mathcal{B}}, \mathtt{a}, v, d) \\
 & ((\mathtt{a}\,,\, q_{\mathcal{A}}\,,\, u) & , & (q'_{\mathcal{B}}\,,\, d) & A_{\mathsf{O}} \times (B_{\mathsf{P}} \times \mathfrak{D}) & \\
\mathsf{P} & \downarrow & & & & \text{where} \quad q'_{\mathcal{A}} := \partial_{\mathcal{A}}(q_{\mathcal{A}}, \mathtt{a}, u, d) \\
 & (q'_{\mathcal{A}} & , & q'_{\mathcal{B}}) & A_{\mathsf{P}} \times B_{\mathsf{P}} &
\end{array}
$$

Figure 29: The edges of the graph $\widetilde{G}$ for $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$

C.2. On Positional Strategies for Separation. Consider now non-deterministic parity automata $\mathcal{A}, \mathcal{B} : \Sigma$ such that $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B}) = \emptyset$. Then by Prop. 7.7 there is a winning P-strategy in $\Sigma \vdash \mathcal{A} \otimes \mathcal{B} \multimap \perp\!\!\!\perp$. It follows from Lem. C.2 that P has winning strategy induced by a function

$$
g \quad : \quad Q_{\mathcal{A}} \times Q_{\mathcal{B}} \times \mathbb{B} \times \Sigma \times U \times V \quad \longrightarrow \quad \mathfrak{D}
$$

The game $\Sigma \vdash \mathcal{A} \otimes \mathcal{B} \multimap \perp\!\!\!\perp$ is won by P if $\perp\!\!\!\perp$ goes to state $\mathbb{t}$, since it can not switch back to $\mathbb{f}$. It follows that it is sufficient to have the values of g above with $\perp\!\!\!\perp$ in state $\mathbb{f}$. It follows that P has a winning strategy in $\Sigma \vdash \mathcal{A} \otimes \mathcal{B} \multimap \perp\!\!\!\perp$ induced by a map of the form

$$
h \quad : \quad Q_{\mathcal{A}} \times Q_{\mathcal{B}} \times \Sigma \times U \times V \quad \longrightarrow \quad \mathfrak{D}
$$

C.3. Proof of Prop. C.1. The proof of Prop. C.1 follows the lines of [CL08], itself based on the complementation construction used in [Tho97, Proof of Thm. 6.9].

Fix a regular $\mathcal{L} \subseteq \Sigma^{\mathfrak{D}^*}$, and consider a non-deterministic parity $\mathcal{C} = (Q_{\mathcal{C}}, q^{\imath}_{\mathcal{C}}, W, \partial_{\mathcal{C}}, \Omega_{\mathcal{C}})$ recognizing the complement of $\mathcal{L}$. Using the closure properties of ω-regular languages, there is a deterministic parity ω-word automaton $\mathcal{D} : \Sigma \times V \times \mathfrak{D}$ where

$$
V \quad := \quad (Q_{\mathcal{C}} \times W \quad \longrightarrow \quad \mathfrak{D})
$$

such that $\mathcal{D}$ accepts $(\mathtt{a}_k, f_k, d_k)_k$ iff for all $(u_k)_k \in U^{\omega}$ and all $(q_k)_k \in Q^{\omega}_{\mathcal{C}}$, we have $(q_k)_k \notin \Omega_{\mathcal{C}}$ whenever $q_0 := q^{\imath}_{\mathcal{C}}$, $q_{k+1} := \partial_{\mathcal{C}}(q_k, \mathtt{a}_k, u_k, f_k(q_k, \mathtt{a}_k, u_k))$, and $d_k = f_k(q_k, \mathtt{a}_k, u_k)$.

Write $\mathcal{D} := (Q_{\mathcal{D}}, q^{\imath}_{\mathcal{D}}, \Omega_{\mathcal{D}})$. Let now $\mathcal{B} : \Sigma$ be a parity non-deterministic automaton with P-moves V and such that an infinite play $((\mathtt{a}_k, f_k) \cdot d_k)_k$ is winning iff $(\mathtt{a}_k, f_k, d_k)_k$ is accepted by $\mathcal{D}$. Explicitly, we let

$$
\mathcal{B} \quad = \quad (Q_{\mathcal{D}}, q^{\imath}_{\mathcal{D}}, V, \partial_{\mathcal{B}}, \Omega_{\mathcal{D}})
$$

where

$$
\partial_{\mathcal{B}}(q, \mathtt{a}, f, d) \quad := \quad \partial_{\mathcal{D}}(q, (\mathtt{a}, f, d))
$$

Lemma C.3 ([Tho97])**.** $\mathcal{L}(\mathcal{B}) = \mathcal{L}$.

Proof of the Lemma. We show that $\mathcal{L}(\mathcal{B}) = \mathcal{L}(\mathcal{C}^{\perp\!\!\!\perp})$. Let $T : \mathfrak{D}^* \to \Sigma$. Assume first that $T \in \mathcal{L}(\mathcal{C}^{\perp\!\!\!\perp})$, so that P has winning strategy in $\mathcal{C}^{\perp\!\!\!\perp}(T)$. Since $\mathcal{C}$ is a parity automaton, this strategy can be assumed to be positional, hence to be determined by a function $\mathfrak{D}^* \to (Q_{\mathcal{C}} \times W \to \mathfrak{D})$. But this determines a P-strategy in $\mathcal{B}(T)$, which is winning by definition of $\mathcal{B}$. Conversely, assume that $T \in \mathcal{L}(\mathcal{B})$. Since $\mathcal{B}$ is non-deterministic, a winning P-strategy in $\mathcal{B}(T)$ is given by a function $\mathfrak{D}^* \to V = \mathfrak{D}^* \to (Q_{\mathcal{C}} \times W \to \mathfrak{D})$. $\square$

Going back to the proof of Prop. C.1, consider a non-deterministic parity $\mathcal{A} : \Sigma$ with $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}$. Since $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{C}) = \emptyset$, it follows from §C.2 that there is a function

$$g \quad : \quad Q_{\mathcal{A}} \times Q_{\mathcal{C}} \times \Sigma \times U \times W \quad \longrightarrow \quad \mathfrak{D}$$

which generates a winning P-strategy in $\Sigma \vdash \mathcal{A} \otimes \mathcal{C} \multimap \perp\!\!\!\perp$. But g can be seen as a map

$$Q_{\mathcal{A}} \times \Sigma \times U \quad \longrightarrow \quad V$$

and this map generates a winning P-strategy in $\Sigma \vdash \mathcal{A} \multimap \mathcal{B}$. □

C.4. A Separation Property from [SA05]. Our internalized linear arrow can handle a construction for the separation property of [SA05, Thm. 2.7].

Consider non-deterministic parity automata $\mathcal{A}, \mathcal{B} : \Sigma$ such that $\mathcal{L}(\mathcal{A}) \cap \mathcal{L}(\mathcal{B}) = \emptyset$. Assume moreover that both $\mathcal{A}$ and $\mathcal{B}$ are parity with colorings of range $\{0, \ldots, n\}$ for some even n. Theorem 2.7 of [SA05] say that there is a parity automaton $\mathcal{C}$ such that $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{C}) \subseteq \mathcal{L}(\mathcal{B}^{\perp\!\!\!\perp})$ and such that $\Omega_{\mathcal{C}}$ is generated by a coloring $c_{\mathcal{B}} : Q_{\mathcal{C}} \to \mathbb{N}$ of range $\subseteq \{0, \ldots, n\}$ and such that in each reachable strongly connected component of $\mathcal{C}$ (for $q \to q'$ iff $q' = \partial_{\mathcal{C}}(q, \mathsf{a}, f, v, d)$ for some a, f, v, d), $c_{\mathcal{C}}$ has range either $\{1, \ldots, n\}$ or $\{0, \ldots, n-1\}$.

We build $\mathcal{C}$ by restricting $\mathcal{B} \multimap \mathcal{A}$ along a winning strategy in $\Sigma \vdash \mathcal{A} \otimes \mathcal{B} \multimap \perp\!\!\!\perp$. By §C.2, there is a function

$$g \quad : \quad Q_{\mathcal{A}} \times Q_{\mathcal{B}} \times \Sigma \times U \times V \quad \longrightarrow \quad \mathfrak{D}$$

which generates a winning P-strategy in $\Sigma \vdash \mathcal{A} \otimes \mathcal{B} \multimap \perp\!\!\!\perp$.

We restrict the automaton $\mathcal{B} \multimap \mathcal{A} : \Sigma$ along g as follows. Recall that $Q_{\mathcal{B} \multimap \mathcal{A}} = Q_{\mathcal{B}} \times Q_{\mathcal{A}}$. Define $\mathcal{C} : \Sigma$ as follows:

$$\mathcal{C} \;:=\; (Q_{\mathcal{B} \multimap \mathcal{A}} + \{\mathbb{t}\}, q^{\imath}_{\mathcal{B} \multimap \mathcal{A}}, U^V, V, \partial_{\mathcal{C}}, \Omega_{\mathcal{C}})$$

where $\partial_{\mathcal{C}}(\mathbb{t}, _, _, _, _) := \mathbb{t}$, and

$$\partial_{\mathcal{C}}((q_{\mathcal{B}}, q_{\mathcal{A}}), \mathsf{a}, f, v, d) \quad := \quad \begin{cases} \mathbb{t} & \text{if } g(q_{\mathcal{A}}, q_{\mathcal{B}}, \mathsf{a}, f(v), v) \neq d \\ \partial_{\mathcal{B} \multimap \mathcal{A}}((q_{\mathcal{A}}, q_{\mathcal{B}}), \mathsf{a}, f, v, d) & \text{otherwise} \end{cases}$$

The coloring $c_{\mathcal{C}}$ of $\mathcal{C}$ is then defined as in [SA05, §2.2.2]. We define it explicitly as follows. Consider a reachable strongly connected component C of $\mathcal{C}$. Note that if C contains $\mathbb{t}$, then $C = \{\mathbb{t}\}$, and we put $c_{\mathcal{C}}(\mathbb{t}) := n$. Otherwise, C contains only states of $\mathcal{B} \multimap \mathcal{A}$, that is states in $Q_{\mathcal{B}} \times Q_{\mathcal{A}}$. Assume that C is non-trivial and contains two states $(_, q_{\mathcal{A}})$ and $(q_{\mathcal{B}}, _)$ with $c_{\mathcal{A}}(q_{\mathcal{A}}) = c_{\mathcal{B}}(q_{\mathcal{B}}) = n$. By definition of $\partial_{\mathcal{C}}$, the set of states

$$\{(q'_{\mathcal{A}}, q'_{\mathcal{B}}, \mathbb{f}) \mid (q'_{\mathcal{B}}, q'_{\mathcal{A}}) \in C\}$$

is reached infinitely often in an infinite play of the strategy in $\Sigma \vdash \mathcal{A} \otimes \mathcal{B} \multimap \perp\!\!\!\perp$ induced by g. But this contradicts the fact that this strategy is winning. It follows that either (a) $c_{\mathcal{A}}$ never takes the value n in C or (b) $c_{\mathcal{B}}$ never takes the value n in C. In the case (a), for each state $(q_{\mathcal{B}}, q_{\mathcal{A}})$ of C we put $c_{\mathcal{C}}(q_{\mathcal{B}}, q_{\mathcal{A}}) := c_{\mathcal{A}}(q_{\mathcal{A}})$, and in the case (b) we put $c_{\mathcal{C}}(q_{\mathcal{B}}, q_{\mathcal{A}}) := c_{\mathcal{B}}(q_{\mathcal{B}}) + 1$.

Consider now an infinite sequence of the form $\rho := (q'_k, q_k)_k \in Q^{\omega}_{\mathcal{B} \multimap \mathcal{A}}$ and let C be a strongly connected component of $\mathcal{C}$ such that $\mathrm{Inf}_k(q'_k, q_k) \subseteq C$. Let $m = \max(\mathrm{Inf}_k c_{\mathcal{C}}(q'_k, q_k))$.

Claim. If m is even, then $\rho \in \Omega_{\mathcal{B} \multimap \mathcal{A}}$

Proof of the Claim. In case (a) above, we have $m = \max(\mathrm{Inf}_k c_{\mathcal{A}}(q_k))$ hence $(q_k)_k \in \Omega_{\mathcal{A}}$ and $\rho \in \Omega_{\mathcal{B} \multimap \mathcal{A}}$. In case (b), $m = \max((\mathrm{Inf}_k c_{\mathcal{B}}(q'_k) + 1))$, hence $\max(\mathrm{Inf}_k c_{\mathcal{B}}(q'_k))$ is odd, so that $(q'_k)_k \notin \Omega_{\mathcal{B}}$ and $\rho \in \Omega_{\mathcal{B} \multimap \mathcal{A}}$. □

Lemma C.4. $\mathcal{L}(\mathcal{C}) \subseteq \mathcal{L}(\mathcal{B}^{\perp\!\!\!\perp})$.

Proof. Consider a winning P-strategy σ in $\mathcal{C}(T)$. Recall that the P-moves of $\mathcal{B}^{\perp\!\!\!\perp}$ are $\mathfrak{D}^V$ and that its O-moves are V, and that the P-moves of $\mathcal{C}$ are U^V and that its O-moves are V. Let τ be the winning P-strategy τ on $\mathcal{A} \otimes \mathcal{B} \multimap \perp\!\!\!\perp$ (whose P-moves are $\mathfrak{D}$ and O-moves are $U \times V$) induced by g. We define a P-strategy θ by combining σ and τ as follows: modulo Currying, θ plays from $v \in V$ the tree direction $d \in \mathfrak{D}$ proposed by $T^\star(\tau)$ from v and the $u \in U$ given by σ on v. Hence the strategies σ and θ play the same moves in $\mathcal{B}$ (provided by O). So the sequences of $Q_{\mathcal{B}}$-states produced by σ and θ are the same, unless O plays in $\mathcal{B}^{\perp\!\!\!\perp}$ a tree direction $d \in \mathfrak{D}$ different from the one proposed by θ, *i.e.* different from the one proposed by τ. In this case, the play on $\mathcal{B}^{\perp\!\!\!\perp}(T)$ is P-winning and we are done. Assume now that the sequences of $Q_{\mathcal{B}}$-states agree. We show that they can not be in $\Omega_{\mathcal{B}}$. Assume toward a contradiction that they are. By the claim above, since σ is winning, the sequence of states in $\mathcal{C}$ belongs to $\Omega_{\mathcal{B} \multimap \mathcal{A}}$ The play respects σ, so the sequence of $Q_{\mathcal{A}}$-states must belong to $\Omega_{\mathcal{A}}$ since σ is winning. But the play also respects $T^\star(\tau)$, which is winning in $\mathcal{A}(T) \otimes \mathcal{B}(T) \multimap \perp\!\!\!\perp$, so the sequence of $Q_{\mathcal{A}}$-states can not belong to $\Omega_{\mathcal{A}}$. It follows that the sequence of $Q_{\mathcal{B}}$-states can not belong to $\Omega_{\mathcal{B}}$, and we are done since the play in $\mathcal{B}^{\perp\!\!\!\perp}(T)$ is then P-winning. □

In order to complete the proof of the separation property, it remains to show the following

Lemma C.5. $\mathcal{L}(\mathcal{A}) \subseteq \mathcal{L}(\mathcal{C})$.

Proof. Let $T : \mathfrak{D}^* \to \Sigma$ such that $T \in \mathcal{L}(\mathcal{A})$. Consider a winning positional P-strategy τ in $\mathcal{A}(T)$ induced by a function $\mathfrak{D}^* \to (Q_{\mathcal{A}} \to U)$. This gives a function $\mathfrak{D}^* \to (Q_{\mathcal{C}} \times V \to U)$ which induces a strategy σ in $\mathcal{C}(T)$. Consider an infinite play ϖ of σ induced by an infinite play ϖ_τ of τ. Let $\rho \in Q_{\mathcal{C}}^\omega$ be the sequence of states produced by ϖ. If ρ contains $\mathbb{t}$, then $\rho \in Q^*_{\mathcal{B} \multimap \mathcal{A}}.\mathbb{t}^\omega \subseteq \Omega_{\mathcal{C}}$ and we are done. Otherwise, let $\rho = (q'_k, q_k)_k \in Q_{\mathcal{B} \multimap \mathcal{A}}$. If we are in case (a) above, then $\max(\mathrm{Inf}_k(c_{\mathcal{C}}(\rho))) = \max(\mathrm{Inf}_k(c_{\mathcal{A}}(q_k)))$, hence $\rho \in \Omega_{\mathcal{C}}$. Assume that we are in case (b), so that $\max(\mathrm{Inf}_k(c_{\mathcal{C}}(\rho))) = \max(\mathrm{Inf}_k(c_{\mathcal{B}}(q'_k)) + 1)$. Let θ be the winning P-strategy in $\Sigma \vdash \mathcal{A} \otimes \mathcal{B} \multimap \perp\!\!\!\perp$ induced by g. Then, by combining ϖ_τ and $\varpi_{\restriction \mathcal{B}}$, we obtain an infinite play ϖ' of θ. Note that in this play, $\perp\!\!\!\perp$ never switches to $\mathbb{t}$ since we assumed $\rho \in \Omega_{\mathcal{B} \multimap \mathcal{A}}$. It follows that ϖ' produces the same sequence of states $(q'_k)_k \in Q_{\mathcal{B}}$ as ϖ, and we must have $(q'_k)_k \notin \Omega_{\mathcal{B}}$ since $(q_k)_k \in \Omega_{\mathcal{A}}$. It follows that $\max(\mathrm{Inf}_k(c_{\mathcal{C}}(\rho))) = \max(\mathrm{Inf}_k(c_{\mathcal{B}}(q'_k)) + 1)$ is even. □

Contents

1. Introduction **1**
- 1.1. MSO and (Non-Deterministic) Tree Automata. . . . 3
- 1.2. Games and Alternating Automata. . . . 3
- 1.3. Toward Linear Logic. . . . 5
- 1.4. Computational Interpretation of Proofs. . . . 6
- 1.5. Toward Realizability Interpretations of MSO. . . . 7
- 1.6. Outline. . . . 10

2. Toward Categories of Games and Automata **10**
- 2.1. Compositionality and Categorical Semantics. . . . 12
- 2.2. Indexed Structure: Substitution and Quantification Rules. . . . 12
- 2.3. Toward a Semantics for Implications. . . . 16
- 2.4. The (Synchronous) Direct Product of (Non-Deterministic) Automata. . . . 17
- 2.5. Alternating Automata and Linear Logic. . . . 19

3. Uniform Automata and Zig-Zag Strategies **20**
- 3.1. Uniform Automata. . . . 20
- 3.2. Full Positive Games and Acceptance for Uniform Automata. . . . 21
- 3.3. Substituted Acceptance Games. . . . 23
- 3.4. Zig-Zag Strategies. . . . 24
- 3.5. Toward Uniform Linear Synchronous Arrow Games. . . . 25

4. Fibrations of Tree Automata **26**
- 4.1. Symmetric Monoidal Structure of $\mathbf{DZ}$ . . . 27
- 4.2. Monoid and Comonoid Indexing in $\mathbf{DZ}$ . . . 29
- 4.3. The Indexed Structure of $\mathsf{DialZ}(-)$ and the Base Category $\mathbf{T}$ . . . 31
- 4.4. The Fibred Category $\mathsf{DialAut}$ . . . 36
- 4.5. Substitution and Language Inclusion . . . 40

5. Symmetric Monoidal Closed Structure **40**
- 5.1. The Symmetric Monoidal Closure of $\mathbf{DZ}$ . . . 41
- 5.2. The Symmetric Monoidal Closed Structure of $\mathsf{DialAut}$ and Tree Automata . . . 42
  - 5.2.1. The Symmetric Monoidal Structure of DialZ. . . . 42
  - 5.2.2. The Symmetric Monoidal Closure of $\mathbf{DZ}_{\mathfrak{D}}$ and DialZ. . . . 43
  - 5.2.3. The Symmetric Monoidal Closed Structure of $\mathsf{DialAut}$. . . . 44
  - 5.2.4. The Symmetric Monoidal Closed Structure of Uniform Automata. . . . 44
- 5.3. Deduction, Adequacy and Correctness . . . 45
- 5.4. Falsity and Complementation . . . 48
  - 5.4.1. Deduction Rules for $\perp$ and $\mathcal{A}^{\perp}$. . . . 49

6. Quantifications **50**
- 6.1. Quantifications in $\mathsf{DialAut}$ . . . 50
- 6.2. Quantifications on Uniform Automata . . . 52
- 6.3. Deduction Rules for Quantifications . . . 53

7. Non-Deterministic Automata **55**
7.1. The Cartesian Structure of Non-Deterministic Automata . . . 55
7.1.1. Application: Deduction Rules for Non-Deterministic Automata. . . . 56
7.1.2. Application: Existential Quantifications and Extraction. . . . 56
7.1.3. Application: Effective Realizers from Witnesses of Non-Emptiness. . . . 58
7.2. Simulation and the Exponential Modality of IMELL . . . 60
7.2.1. Parity Automata. . . . 61
7.2.2. An Exponential Construction on Uniform Automata. . . . 62
7.2.3. Game Graphs and Positionality. . . . 63
7.2.4. Applications. . . . 64

8. Conclusion **65**
8.1. Further Works. . . . 66

A. Linear Synchronous Arrow Games **73**
A.1. Simple Games. . . . 73
A.2. Game Semantics: Linear Arrow Games and Copy-Cat. . . . 76
A.3. Uniform Linear Synchronous Arrow Games. . . . 77
A.4. Zig-Zag Games. . . . 78
A.5. The Category $\mathbf{DZ}^{(W)}$ of Zig-Zag Games and Total (Winning) Strategies. . . . 81

B. Proof of Adequacy of the Promotion Rule (Prop. 7.13) **84**

C. Further Examples **86**
C.1. On Positional Strategies. . . . 86
C.2. On Positional Strategies for Separation. . . . 87
C.3. Proof of Prop. C.1. . . . 87
C.4. A Separation Property from [SA05]. . . . 88